

---

# 1 Einleitung

Dieses Handbuch beschreibt alle Makros und Exits, die für die Arbeit mit den Spool & Print Services von BS2000/OSD zur Verfügung stehen.

## 1.1 Kurzbeschreibung der Spool & Print Services

Die Spool & Print Services für BS2000/OSD bestehen aus verschiedenen Subsystemen und Dienstprogrammen.

### Spool & Print Subsysteme

#### SPOOL

Der lokale SPOOL (**S**imultaneous **P**eripheral **O**peration **O**n-**L**ine) ist ein BS2000-Subsystem. Er steuert die asynchrone Ausgabe auf Drucker und Magnetbänder. SPOOL ist die grundlegende Komponente für die anderen, nachfolgend beschriebenen Produkte, die ohne SPOOL nicht ablaufen können. Zur Beschreibung siehe Handbuch „[SPOOL \(BS2000/OSD\)](#)“.

#### RSO

**R**emote **S**pool **O**utput steuert die Ausgabe auf dezentrale Drucker (RSO-Drucker), die an fernen Rechnern angeschlossen sind. RSO lässt nur einen Punkt-zu-Punkt-Transfer der Daten zu. Zur Beschreibung siehe Handbuch „[RSO \(BS2000/OSD\)](#)“.

#### Dprint

**D**istributed **P**rint Services steuert ebenfalls die Ausgabe auf Drucker, die an fernen Rechnern angeschlossen sind, benutzt dazu aber den lokalen SPOOL des Zielrechners und behält die Kontrolle über die Druckausgabe. Zur Beschreibung siehe Handbuch „[Distributed Print Services \(BS2000/OSD\)](#)“.

#### SPS

**S**POOL**A**P**A** **P**rinting **S**ystem kann als Subsystem an das SPOOL-System angebunden werden und nimmt die Aufgabe eines Druckertreibers für APA-Drucker wahr. Zur Beschreibung siehe Handbuch „[PRISMAproduction/BS2000 - SPS](#)“.

## IDOM

Integrated **D**ocument and **O**utput **M**anagement ist ein Subsystem in Spool & Print Services auf BS2000/OSD. Es erweitert die Spool & Print Services um die Möglichkeit des Dokumenten-Managements. Zur Beschreibung siehe Handbuch „[IDOM \(BS2000/OSD\)](#)“.

## Zusätzliche Dienstprogramme zur Verwaltung

### PRM

Das **P**rint **R**esource **M**anagement dient zur Erstellung und Verwaltung von SPOOL-Druckressourcen. PRM ist obligatorisch ab der SPOOL-Version 3.0, da alle Aktionen bezüglich der Druckressourcen über PRM abgewickelt werden. PRM wird im gleichnamigen Handbuch beschrieben.

### SNS

Der **S**POOL **N**otification **S**ervice dient zum Senden und Verwalten von Benachrichtigungen im Rahmen von BS2000.

### SPSERVE

Mit SPSEVERE kann die Systembetreuung - in eingeschränktem Umfang auch der RSO-Geräteverwalter und der nichtprivilegierte Benutzer - Informationen in einer SPOOL-Parameterdatei eintragen, ändern, löschen oder sich ausgeben lassen. SPSEVERE wird im gleichnamigen Handbuch beschrieben.

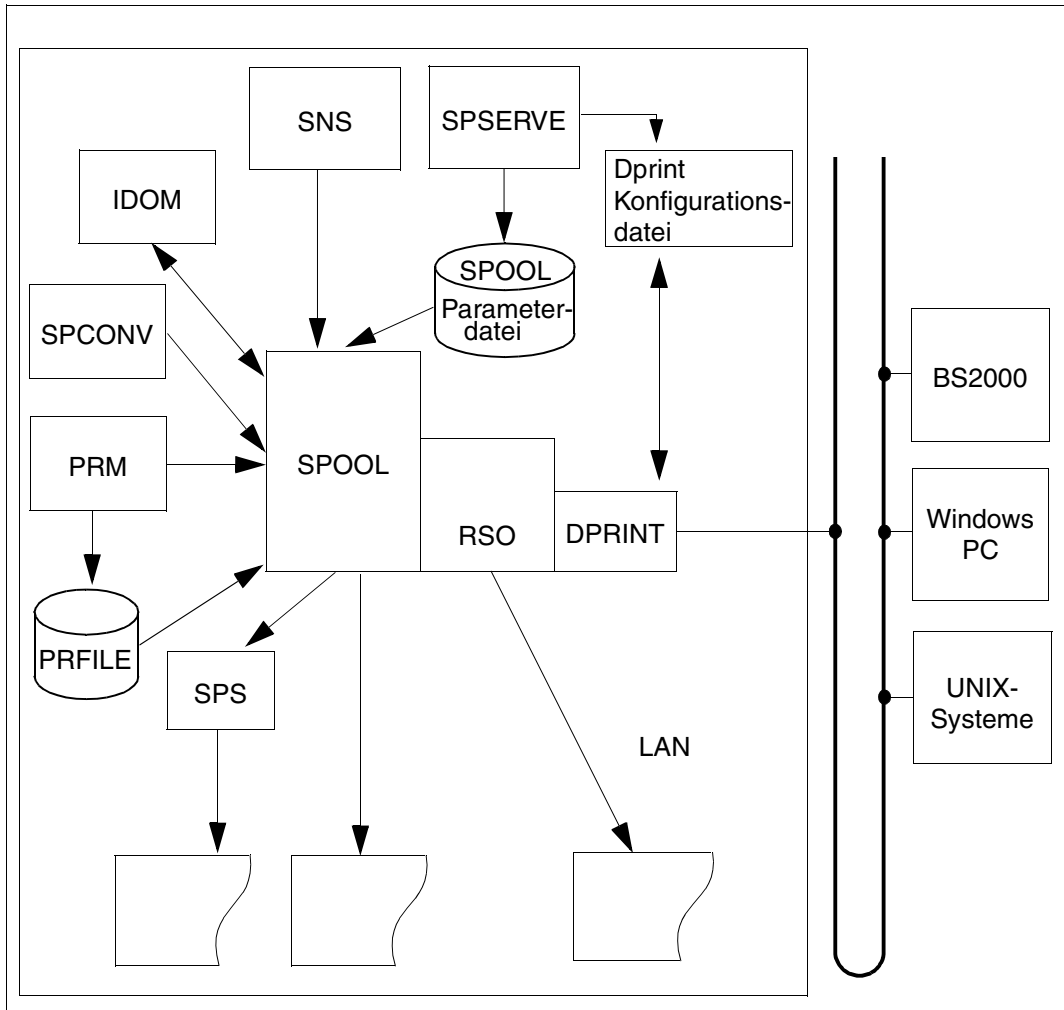
### SPCONV

Der SPOOL-CONVERTER realisiert die Filtertechnik für das Drucken in verteilten, heterogenen Umgebungen mit BS2000-, UNIX- und PC-Systemen, d.h. er bringt Dokumente und Druckressourcen durch Filter in ein Format, das von den jeweiligen Druckern ausgedruckt werden kann. Zu SPCONV siehe das gleichnamige Handbuch.

### Spool&Print Explorer

Der Spool&Print Explorer ist die grafische Benutzerschnittstelle (GUI) für die Spool & Print -Verwaltung auf BS2000-Systemen. Der Spool&Print Explorer ist ein zentrales und grafisch orientiertes Administrationsprogramm auf PC für Spool & Print -Verwalter und/oder -Endbenutzer (siehe Handbuch "[SPOOL \(BS2000/OSD\)](#)").

Das folgende Bild zeigt die Bestandteile der Spool & Print Services



## 1.2 Zielgruppe des Handbuchs

Dieses Handbuch wendet sich an

- die BS2000-Systembetreuung (Exits)
- an den Benutzer, der mit Hilfe der Programmierschnittstellen Dateien drucken oder auf externe Datenträger schreiben will.
- den SPOOL-Verwalter (ab BS2000/OSD V2.0)
- den RSO-Geräteverwalter
- den nichtprivilegierten Benutzer der Spool & Print Services, der seine Daten auf BS2000-Hochleistungsdruckern ausgeben will.

## 1.3 Aufbau des Handbuchs

Die Beschreibungen in diesem Handbuch sind innerhalb von Sachgebieten alphabetisch sortiert. Am Beginn jedes Sachgebiets finden Sie eine kurze Beschreibung seines Inhalts.

### Kapitel „Makros für die Ausgabe von Dateien“

enthält die Beschreibung der Makros, mit denen Daten ausgegeben und Druckaufträge vergeben werden.

### Kapitel „Makros für die Ausgabe von Informationen“

enthält die Beschreibung des Makros, mit dem der Inhalt der SPOOL-Parameterdatei abgefragt werden kann.

### Kapitel „Makros für die Verwaltung von Druckaufträgen“

enthält die Beschreibung der Makros, mit denen Druckaufträge verwaltet werden.

### Kapitel „Makros für die Verwaltung von Druckern“

enthält die Beschreibung der Makros, mit denen Drucker verwaltet werden.

### Kapitel „Makros für virtuelle Drucker“

beschreibt Anwendungsmodelle für virtuelle Drucker und die Makros, mit denen virtuelle Drucker genutzt werden können.

### Kapitel „Spool-Exits“

enthält die Beschreibung der Exits der Spool & Print Services.

Im **Anhang** befinden sich folgende Abschnitte:

- „**Makrosyntax**“ mit einer Beschreibung der in diesem Handbuch benutzten Makrosyntax.
- „**Funktionsweise des Exit-Mechanismus**“ beschreibt die Funktion der Exits.

- „**Schlussblatt für lokalen SPOOL**“ mit einem Abdruck des Schlussblattes für einen Druckauftrag des lokalen SPOOLS.
- „**SPOOLOUT-Abrechnungssatz SPLO**“ mit der Beschreibung des Abrechnungssatzes des SPOOLS.
- „**Anwendungsregeln für die Sprache C**“ mit der Beschreibung der Anwendungsregeln für die Sprache C.

Am Ende des Handbuchs finden Sie das Literaturverzeichnis und das Stichwortverzeichnis.

## 1.4 Änderungen der Makros und Exits gegenüber den Vorgängerversionen

Allgemeine Anpassungen an Distributed Print Services V1.1B, RSO V3.4., SNS V1.0B, SPOOL V4.6, SPS V3.8 und SPSERVE V2.8 wurden in das Handbuch eingefügt. Folgende Änderungen/Erweiterungen wurden an der Programm-Schnittstelle vorgenommen:

- In den Makros SASDSPO, STRSPPR und SUPSPPR wurde für einige \*\_TYP-Parameter ein Wert \*CLIST hinzugefügt.
- Die korrekte Syntaxbeschreibung der Makros SPVDRCV und SPVDRET ist aufgeführt.
- Zum Makro SHOPJST wurde ein neuer Returncode hinzugefügt.
- Das neue Makro SNPPRNT beschreibt die Parameterliste für die Unterstützung des Notification Service.
- Das Makro PRNTDOC wurde erweitert. Der Operand PRNAME wurde hinzugefügt, damit eine URL angegeben werden kann und der Operand NOTIFPAR wurde hinzugefügt, um den Zugriff zum Notification Service per Programm zu ermöglichen.
- Die Makros PRNTRSO, MODJAT und MODJRSO wurden erweitert, damit eine URL angegeben werden kann.
- Das neue Makro SSVBARC zur Bar Code-Unterstützung wurde hinzugefügt.
- Die folgenden Operanden der Makros PRNTDOC und MODPJAT gelten jetzt auch für APA-Drucker: INTRAY, LEFTMAR, LEFTOFF, OUTTRAY, ROT, TOPOFF, TWOSIDE
- Zwei neue Operanden wurden zu den Makros PRNTSPS und MPJASPS hinzugefügt: FRONTOVL und BACKOVL
- Das neue Makro OLTPRQX bietet UTM-Anwendungen die Möglichkeit, beim Ausdruck von Nachrichten verschiedene Druckparameter einzustellen.

- Abrechnung:  
Um eine Synchronisierung zwischen SPOOL V4.4 und SPS 03.08.xy sicherzustellen, wurde die Erweiterung für APA-Seitendrucker des SPOOL-Abrechnungssatzes geändert. Alle anderen Felder der SPOOL-Abrechnung bleiben unverändert.
- Ab BS2000/OSD V4.0 werden Diskettengeräte nicht mehr unterstützt, daher wird das Makro PNCH nicht mehr in diesem Handbuch beschrieben.
- SCSIPL  
Das Produkt SCSIPL wird ab SPOOL V4.3 nicht mehr unterstützt.

## 1.5 Verwendete Darstellungsmittel



Dieses Symbol weist daraufhin, dass der folgende eingerückte Absatz unabdingbare Informationen enthält.

### *Hinweis*

Das Wort „*Hinweis*“ vor einem eingerückten Absatz zeigt an, dass der folgende Absatz wichtige Informationen enthält.

### „Verweis“

Verweise auf Kapitel, Abschnitte oder andere Handbücher sind in Anführungszeichen eingeschlossen.

[ ]

Eckige Klammern in Syntaxdarstellungen: Die Zeichen innerhalb der Klammern dürfen weggelassen werden.

### **Fettdruck**

Wo Syntaxdarstellungen erläutert werden, sind die Zeilen, die aktuell erläutert werden, halbfett dargestellt.

Im Übrigen gelten für Syntaxdarstellungen die Regeln, wie sie in den entsprechenden Kapiteln des Nachschlageteils beschrieben sind.

### SYNTAX/Beispiel

Syntaxdarstellungen und Beispiel-Eingaben und -Ausgaben werden durch andere Schriften hervorgehoben. Syntaxdarstellungen sind außerdem von einem Rahmen umgeben.

## 1.6 Readme-Datei

Funktionelle Änderungen und Nachträge der aktuellen Produktversionen zu diesem Handbuch entnehmen Sie bitte ggf. den produktspezifischen Readme-Dateien.

Sie finden die Readme-Datei auf Ihrem BS2000-Rechner unter dem Dateinamen SYSRME.produkt.version.D.

Die Benutzerkennung, unter der sich die Readme-Dateien befinden, erfragen Sie bitte bei Ihrer zuständigen Systembetreuung. Die Readme-Dateien können Sie mit dem Kommando /SHOW-FILE oder mit einem Editor ansehen oder auf einem Standarddrucker mit folgendem Kommando ausdrucken:

```
/PRINT-DOCUMENT FROM-FILE=dateiname, -  
/DOCUMENT-FORMAT=*TEXT(LINE-SPACING=*BY-EBCDIC-CONTROL)
```





---

## 2 Makros für die Ausgabe von Dateien

In diesem Kapitel sind die folgenden Makros beschrieben:

[„OLTPRQX - UTM-Druckparameter festlegen“ auf Seite 10](#)

Das Makro OLTPRQX bietet UTM-Anwendungen die Möglichkeit, beim Ausdruck von Nachrichten verschiedene Druckparameter einzustellen.

[„PRNTDOC \(für SPOOL, SPS, RSO und Dprint\) - Datei ausgeben“ auf Seite 29](#)

Das Makro PRNTDOC gibt Dateien auf Druckern und anderen Peripheriegeräten aus. Diese Geräte müssen nicht am lokalen Rechner angeschlossen sein, da PRNTDOC bei passender Einstellung der Parameter auch mit RSO und Dprint zusammenarbeitet.

[„PRNTRSO - RSO-Parameter für PRNTDOC-Makro festlegen“ auf Seite 176](#)

Wenn PRNTDOC Dateien via RSO auf fernen Systemen ausgeben soll, müssen Sie mit PRNTRSO die entsprechenden Parameter passend einstellen.

[„PRNTDPC - Dprint-Parameter für PRNTDOC-Makro definieren“ auf Seite 182](#)

Sobald PRNTDOC Dateien via Dprint auf fernen Systemen ausgeben soll, müssen Sie mit PRNTDPC die entsprechenden Parameter passend einstellen.

[„PRNTSPS - SPS-APA-Parameter für PRNTDOC-Makro definieren“ auf Seite 187](#)

Sobald PRNTDOC Dateien auf HP-Druckern ausgeben soll, müssen Sie mit PRNTSPS die entsprechenden Parameter passend einstellen.

[„SNPPRNT - Notification-Parameter für PRNTDOC definieren“ auf Seite 192](#)

Das Makro SNPPRNT beschreibt die Parameterliste für die Unterstützung des Notification Service.

[„SSVBARC - Datenkonvertierung für Bar Codes“ auf Seite 197](#)

Das Makro SSVBARC stellt einen Standardweg für den Aufruf des Bar Code-Programms zur Verfügung. Das Bar Code-Programm (LLM) konvertiert ein Zeichen in die entsprechende Darstellung als Bar Code.

## OLTPRQX - UTM-Druckparameter festlegen

**Benutzergruppe:** UTM-Anwendungen

**Programmiersprachen:** Assembler, CPP, C, Cobol

**Makrotyp:** M, S

Der Makro OLTPRQX bietet UTM-Anwendungen die Möglichkeit, beim Ausdruck von Nachrichten Folgendes einzustellen:

- Auswahl der Orientierung: Portrait, Landscape, Inverse Portrait und Inverse Landscape
- Auswahl der Papierquelle durch den Papiertyp (A4, A3 etc.) oder durch die Nummer des Eingabefachs.
- Auswahl des Ausgabefachs
- Auswahl der Druckgestaltung: Offset der logischen Seite auf dem Medium, Simplex-, Duplex- oder Tumble-Duplex-Druck, Anzahl der Seitenkopien.

### Format

Operation	Operanden
OLTPRQX	<pre>,INTRAY=*IGNORE / &lt;integer 1..99&gt; / *BY-FORMAT ,INTRAYF=*C5 / *DL / *COMMERCIAL_10 / *MONARCH / *DOUBLE-LETTER / *LETTER / *LEGAL / *EXEC / *INVOICE / *FOLIO / *B5 / *B4 / *A5 / *A4 / *A3 / *MANUAL ,LOFSIGN=*LOFSPOS / *LOFSNEG ,LOFVAL=*IGNORE / &lt;integer 1..255&gt; ,OUTTRAY=*IGNORE / &lt;integer 1..99&gt; / *SORTER ,PAGECOP=*IGNORE / &lt;c_string 3..3&gt; ,ROT=*NO / *ROT_0 / *ROT_90 / *ROT_180 / *ROT_270 ,SRTMODE=*NO / *NO_SORTER / *GROUP / *COLLATE / *STACKER / *AUTOMATIC ,TOFSIGN=*TOPSPOS / *TOPSNEG ,TOFVAL=*IGNORE / &lt;integer 1..255&gt; ,TWOSIDE=*NO / *SIMPLEX / *DUPLEX / *TUMBLE</pre>

## Operandenbeschreibung

### **INTRAY=\*IGNORE / <integer 1..99> / \*BY-FORMAT**

Papier-Eingabefach für die RSO-Drucker 2030-PCL, 4011, 4812, 4813, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9000-EPLQ, 9000-EPsq, 9004, 9011, 9012, 9013, 9014, 9015, 9021, 9022, 9022-200, 9025, 9026-PCL, 9026-RENO, 9097 und DJET.

### **INTRAY=\*IGNORE**

Die Voreinstellung des Druckers wird verwendet.

### **INTRAY=<integer 1..99>**

Nummer des Eingabefachs, aus dem das Papier sowohl für den Druck der Datei selbst als auch für Deck- und Schlussblatt entnommen wird. Der angegebene Wert wird von RSO nicht überprüft. Werte zwischen 1 und 99 sind möglich. Weitere Informationen über die möglichen Werte, die bei den einzelnen Druckertypen angegeben werden können, finden Sie in der Beschreibung des PRINT-DOCUMENT-Kommandos im Handbuch „[Spool & Print - Kommandos \(BS2000/OSD\)](#)“.

### **INTRAY=\*BY-FORMAT**

Das Papier-Eingabefach wird durch das Schlüsselwort für das Format im INTRAYF-Operanden ausgewählt.

### **INTRAYF=\*C5 / \*DL / \*COMMERCIAL\_10 / \*MONARCH / \*DOUBLE-LETTER / \*LETTER / \*LEGAL / \*EXEC / \*INVOICE / \*FOLIO / \*B5 / \*B4 / \*A5 / \*A4 / \*A3 / \*MANUAL**

Wählt das Papier-Eingabefach durch ein Schlüsselwort für das Format aus. Weitere Informationen über die möglichen Werte bei den einzelnen Druckertypen finden Sie in der Beschreibung des PRINT-DOCUMENT-Kommandos im Handbuch „[Spool & Print - Kommandos \(BS2000/OSD\)](#)“.

### **INTRAYF= \*MANUAL**

Bei Angabe des Operanden \*MANUAL wird das Papier manuell eingegeben und Sie müssen daher immer dann ein Blatt eingeben, wenn eine neue Seite ausgedruckt werden soll oder wenn der Drucker Sie dazu auffordert. Folgende Drucker unterstützen den manuellen Papiereinzug: 9014, 9021, 9022, 9022-200, 9026-RENO, 9026-PCL, 4812, DJET, 4814-PCL, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL and 9000-PCL.

### **LOFSIGN=\*LOFSPOS / \*LOFSNEG**

Gibt das Offset-Zeichen an: positiv (\*LOFSPOS) oder negativ (\*LOFSNEG). Bei positivem Wert wird die erste Zeile nach rechts gerückt, bei negativem Wert nach links.

**LOFVAL=\*IGNORE / <integer 1..255>**

Gibt den Abstand (Offset) des linken Randes der Druckseite vom linken Rand der physikalischen Seite in Millimetern an. Zuerst wird die Druckseite auf dem Papier positioniert und anschließend wird der Text innerhalb der Druckseite gedreht (Rotation) und positioniert. Dies bedeutet, dass die Text-orientierung innerhalb der Druckseite nicht berücksichtigt wird, wenn die Druckseite auf dem Papier verschoben wird. Dieser Operand kann nur für die Drucker 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9021, 9022-200, 9026-PCL und 2030-PCL angegeben werden.

**LOFVAL=\*IGNORE**

Die Druckseite wird nicht horizontal auf dem Papier verschoben.

**LOFVAL=<integer 1..255>**

Die Druckseite wird horizontal auf dem Papier um den angegebenen Wert verschoben.

**OUTTRAY=\*IGNORE / <integer 1..99> / \*SORTER**

Gibt das Papier-Ausgabefach an für die RSO-Drucker 2030-PCL, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9014, 9015, 9026-PCL und 9026-RENO.

**OUTTRAY=\*IGNORE**

Die Voreinstellung des Druckers wird verwendet.

**OUTTRAY=<integer 1..99>**

Gibt das Papier-Ausgabefach an, das für den aktuellen Druckauftrag verwendet werden soll. Weitere Informationen über die den Druckertypen zugeordneten Werte finden Sie in der Beschreibung des PRINT-DOCUMENT-Kommandos im Handbuch „[Spool & Print - Kommandos \(BS2000/OSD\)](#)“.

**OUTTRAY=\*SORTER**

Für den aktuellen Druckauftrag soll der im Operanden SRTMODE angegebene Sortier-Mechanismus verwendet werden. Der Sortier-Mechanismus ist zulässig für die Drucker 9026-RENO, 4818-PCL, 4822-PCL und 4825-PCL und kann für bis zu 20 Ausgabefächer angewendet werden. Er kann nicht über die im Geräteeintrag definierten Voreinstellungen gesteuert werden.

**PAGECOP=\*IGNORE / <c\_string 3..3>**

Gibt an, wie oft jede einzelne Seite wiederholt werden soll. Nur für die RSO-Drucker 2030-PCL, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9021, 9022, 9022-200, 9025, 9026-PCL, 9026-RENO und DJET.

**PAGECOP=\*IGNORE**

Die Voreinstellung des Druckers wird verwendet.

**PAGECOP=<c\_string 3..3>**

Anzahl der Seitenkopien. Mögliche Werte '000' .. '255'.

PAGECOP=255 hat den gleichen Effekt wie PAGECOP=254: ein Original und 254 Kopien werden gedruckt.

**ROT=\*NO / \*ROT\_0 / \*ROT\_90 / \*ROT\_180 / \*ROT\_270**

Gibt an, ob die Seiten eines Druckauftrags gedreht ausgedruckt werden sollen und falls ja, um wieviel Grad gedreht.

**ROT=\*NO**

Keine Seitenrotation ist ausgewählt. Weitere Informationen finden Sie in der Beschreibung des PRINT-DOCUMENT-Kommandos im Handbuch „[Spool & Print - Kommandos \(BS2000/OSD\)](#)“.

**ROT=\*ROT\_0 / \*ROT\_90 / \*ROT\_180 / \*ROT\_270**

Jede Druckseite wird um 0° / 90° / 180° / 270° (im Uhrzeigersinn) gedreht und ausgedruckt. Rotation 0/90 wird von den Druckertypen 2030-PCL, 4812, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9021, 9022-200, 9025, 9026-PCL, 9026-RENO und DJET unterstützt.

Rotation 180/270 wird von den Druckertypen 2030-PCL, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9026-PCL und 9026-RENO unterstützt.

**SRTMODE=\*NO / \*NO\_SORTER / \*GROUP / \*COLLATE / \*STACKER / \*AUTOMATIC**

Der Sortier-Mechanismus soll für den aktuellen Druckauftrag angewendet werden. Der Sortier-Mechanismus ist zulässig für die Drucker 9026-RENO, 4818-PCL, 4822-PCL und 4825-PCL und kann für bis zu 20 Ausgabefächer angewendet werden. Er kann nicht über die im Geräteeintrag definierten Voreinstellungen gesteuert werden. Beispiele hierzu finden Sie in der Beschreibung des PRINT-DOCUMENT-Kommandos im Handbuch „[Spool & Print - Kommandos \(BS2000/OSD\)](#)“.

**SRTMODE=\*NO**

Die Voreinstellung des Druckers wird verwendet.

**SRTMODE=\*NO\_SORTER**

Alle Dokumentseiten werden in den Sortierfächern von unten nach oben abgelegt.

Ausnahme 9026-RENO: Hier werden die Druckseiten je nach optimaler Zugänglichkeit in ein Papierfach abgelegt.

**SRTMODE=\*GROUP**

Wird SRTMODE=\*GROUP zusammen mit PAGECOP angegeben, dann wird jede Kopie einer bestimmten Dokumentseite in ein eigenes Sortierfach abgelegt. Jede folgende Dokumentseite wird auf die vorherigen Seiten abgelegt. Nach Ende des Druckauftrags enthält jedes Sortierfach eine komplette Kopie des gedruckten Dokuments.

**SRTMODE=\*COLLATE**

Wird SRTMODE=\*COLLATE zusammen mit PAGECOP angegeben, dann werden alle Kopien einer bestimmten Dokumentseite in ein Sortierfach abgelegt. Die Kopien der nächsten Dokumentseite werden in das nächste Sortierfach abgelegt. Die Sortierfächer werden von unten nach oben gefüllt.

**SRTMODE=\*STACKER**

Nicht für den Drucker 9026 anwendbar.

Alle gedruckten Seiten werden im Massenausgabefach des Sortiermechanismus abgelegt; und zwar bis zu einem Maximum von 500 Seiten. Dieser Modus bietet sich dafür an, wenn eine einzige Kopie von einem sehr langen Dokument ausgedruckt werden soll

**SRTMODE=\*AUTOMATIC**

Nur für den Drucker 9026 anwendbar.

Abhängig von der in PAGE-COPIES angeforderten Anzahl der Kopien pro Seite und der Anzahl der Sortierfächer wird das Ablageverhalten automatisch gewählt. Die bedruckten Seiten werden entweder unsortiert oder sortiert abgelegt. Die bedruckten Seiten werden wie bei \*NO unsortiert abgelegt, wenn die bei PAGE-COPIES angegebene Anzahl der Kopien gleich der Anzahl der Sortierfächer ist. Die bedruckten Seiten werden wie bei \*GROUP nach Dokumenten sortiert, wenn die bei PAGE-COPIES angegebene Anzahl der Kopien kleiner als die Anzahl der Sortierfächer ist. Die bedruckten Seiten werden wie bei \*COLLATE nach Seiten sortiert in den Sortierfächern abgelegt, wenn die bei PAGE-COPIES angegebene Anzahl der Kopien größer als die Anzahl der Sortierfächer ist.

**TOFSIGN=\*TOPSPOS / \*TOPSNEG**

Gibt das Offset-Zeichen an: positiv (\*TOPSPOS) oder negativ (\*TOPSNEG). Bei positivem Wert wird die erste Zeile nach unten gerückt, bei negativem Wert nach oben.

**TOFVAL=\*IGNORE / <integer 1..255>**

Legt den Abstand des oberen Randes der Druckseite vom oberen Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der Druckseite nicht beachtet wird.

Dieser Operand ist nur für die Drucker 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9021, 9022-200, 9026-PCL und 2030-PCL zulässig.

**TOFVAL=\*IGNORE**

Die Druckseite wird nicht vertikal auf dem Papier verschoben.

**TOFVAL=<integer 1..255>**

Die Druckseite wird vertikal auf dem Papier um den angegebenen Wert verschoben.

**TWOSIDE=\*NO / \*SIMPLEX / \*DUPLEX / \*TUMBLE**

Legt für RSO-Drucker fest, ob das Papier einseitig oder beidseitig bedruckt werden soll.

**TWOSIDE=\*NO**

Die Voreinstellung des Druckers wird verwendet.

**TWOSIDE=\*SIMPLEX**

Der Druckauftrag wird im Simplex-Modus, d.h. einseitig gedruckt.

**TWOSIDE=\*DUPLEX**

Der Druckauftrag wird im Duplex-Modus, d.h. beidseitig gedruckt. Die Seiten werden entlang einer senkrechten Kante gebunden. Duplex-Modus ist erlaubt für die Drucker 2030-PCL, 9026-PCL, 9026-RENO, 4822-PCL, 4825-PCL, 4824-PCL, 4830-PCL, 4850-PCL und 9000-PCL.

**TWOSIDE=\*TUMBLE**

Es wird im Duplex-Modus, d.h. beidseitig, gedruckt, und die Seiten werden nicht von links nach rechts, sondern von oben nach unten umgedreht. Der Modus Tumble-Duplex ist erlaubt für die Drucker 2030-PCL, 9026-PCL, 9026-RENO, 4822-PCL, 4825-PCL, 4824-PCL, 4830-PCL, 4850-PCL und 9000-PCL.

**Struktur-Layouts****ASSEMBLER-Spezifikationen**

```

*   parameter area
OLTXHDR  FHDR  MF=(C,OLTX),EQUATES=NO    header layout
*
OLTXEXT  DS    0XL28                      extension part layout
OLTXR0T  DS    FL1                        rotation code
*   description of the rotation codes
      AIF  ('&EQUATES' NE 'YES').Q01001
OLTXR0T  EQU    0                          no rotation given
OLTXR0T0 EQU   241                        no rotation
OLTXR90  EQU   242                        rotation 90
OLTXR180 EQU   243                        rotation 180
OLTXR270 EQU   244                        rotation 270
.Q01001 ANOP
*
OLTXIT#  DS    0XL1                        intray
OLTXINT  DS    X                          input tray number
      ORG  OLTXIT#
OLTXITFS DS    FL1                        input tray format
*   description of the input tray selection mode
      AIF  ('&EQUATES' NE 'YES').Q01002
OLTXITBF EQU   255                        selection via a format name
OLTXITIG EQU    0                          no input tray given

```

```

.Q01002 ANOP
*
      ORG  OLTXIT#
OLTXTIRA DS  FL1          input tray code
*  description of given input tray
      AIF  ('&EQUATES' NE 'YES').Q01003
OLTXTNTR EQU  0          no input tray given
OLTXTM#  EQU  99        max num tray #
OLTXTC5  EQU  239      x'EF' C5
OLTXTDL  EQU  240      x'F0' DL
OLTXTIC10 EQU  241     x'F1' COM10
OLTXTMRC EQU  242     x'F2' monarch
OLTXTDBL EQU  243     x'F3' double letter
OLTXTLET EQU  244     x'F4' letter
OLTXTLEG EQU  245     x'F5' legal
OLTXTXEC EQU  246     x'F6' exec
OLTXTVCE EQU  247     x'F7' invoice
OLTXTIFOL EQU  248    x'F8' folio
OLTXTB5  EQU  249     x'F9' B5
OLTXTB4  EQU  250     x'FA' B4
OLTXTA5  EQU  251     x'FB' A5
OLTXTA4  EQU  252     x'FC' A4
OLTXTA3  EQU  253     x'FD' A3
OLTXTIMAN EQU  254    x'FE' manual
.Q01003 ANOP
*
      ORG  OLTXT#*+1
OLTXT#   DS  0XL1      outtray
OLTXTOUT DS  X         output tray number
      ORG  OLTXT#
OLTXTTRA DS  FL1      output tray code
*  description of the output tray code
      AIF  ('&EQUATES' NE 'YES').Q01004
OLTXTNTR EQU  0          no output tray
OLTXTM#  EQU  99        output tray max #
OLTXTSR  EQU  129      x'81' otray sorter
.Q01004 ANOP
*
      ORG  OLTXT#*+1
OLTXTSORT DS  FL1      sorter mode
*  description of the sorter codes
      AIF  ('&EQUATES' NE 'YES').Q01005
OLTXTNSRG EQU  0        no sorter given
OLTXTNSRT EQU  241     x'F1' sorter=no
OLTXTSGRP EQU  242     x'F2' sorter=group
OLTXTSCOL EQU  243     x'F3' sorter=collate
OLTXTSSTK EQU  244     x'F4' sorter=stacker
OLTXTSAUT EQU  245     x'F5' sorter=automat

```



```

.Q01005 ANOP
*
OLTXLOFS DS FL1 left offset sign
* description of the sign codes
    AIF ('&EQUATES' NE 'YES').Q01006
OLTXPOSI EQU 78 positive
OLTXNEGA EQU 96 negative
.Q01006 ANOP
*
OLTXLOFT DS X left offset binary; x'00'=no
* left offset
OLTXTOFS DS FL1 top offset sign
OLTXTOFT DS X top offset binary; x'00'= no
* top offset
OLTXDUPX DS FL1 duplex
* description of the twoside codes
    AIF ('&EQUATES' NE 'YES').Q01007
OLTXNDUP EQU 0 no simplex/duplex
OLTXSPLX EQU 241 x'F1' simplex
OLTXDPLX EQU 242 x'F2' duplex
OLTXTDPX EQU 243 x'F3' tumble duplex
.Q01007 ANOP
*
OLTXHCOP DS CL3 hdwr page copies; c'000' or
* first position=x'00' means
* no copies
OLTXRSV2 DS CL16 free for use
*
    AIF ('&EQUATES' NE 'YES').Q01008
OLTX# EQU *-OLTXHDR
.Q01008 ANOP

```

## C++-Spezifikationen

```

/* description of the rotation codes */
/* ENUM _rotation */
#define OLTPRQXNO_ROT_GIVEN 0 /* no rotation given */
#define OLTPRQXROT_0 241 /* no rotation */
#define OLTPRQXROT_90 242 /* rotation 90 */
#define OLTPRQXROT_180 243 /* rotation 180 */
#define OLTPRQXROT_270 244 /* rotation 270 */

/* description of the input tray selection mode */
/* ENUM _intray_sel */
#define OLTPRQXBY_FORMAT 255 /* selection via a format */
/* name */
#define OLTPRQXIGNORE 0 /* no input tray given */

```

```

/* description of given input tray */
/* ENUM _input_tray */
#define OLTPRQXNO_ITRAY 0 /* no input tray given */
#define OLTPRQXI_TRAY_NR 99 /* max num tray # */
#define OLTPRQXI_TRAY_C5 239 /* x'EF' C5 */
#define OLTPRQXI_TRAY_DL 240 /* x'F0' DL */
#define OLTPRQXI_TRAY_COM10 241 /* x'F1' COM10 */
#define OLTPRQXI_TRAY_MONARCH 242 /* x'F2' monarch */
#define OLTPRQXI_TRAY_DOUBLET 243 /* x'F3' double letter */
#define OLTPRQXI_TRAY_LETTER 244 /* x'F4' letter */
#define OLTPRQXI_TRAY_LEGAL 245 /* x'F5' legal */
#define OLTPRQXI_TRAY_EXEC 246 /* x'F6' exec */
#define OLTPRQXI_TRAY_INVOICE 247 /* x'F7' invoice */
#define OLTPRQXI_TRAY_FOLIO 248 /* x'F8' folio */
#define OLTPRQXI_TRAY_B5 249 /* x'F9' B5 */
#define OLTPRQXI_TRAY_B4 250 /* x'FA' B4 */
#define OLTPRQXI_TRAY_A5 251 /* x'FB' A5 */
#define OLTPRQXI_TRAY_A4 252 /* x'FC' A4 */
#define OLTPRQXI_TRAY_A3 253 /* x'FD' A3 */
#define OLTPRQXI_TRAY_MANUAL 254 /* x'FE' manual

/* description of the output tray code */
/* ENUM _output_tray */
#define OLTPRQXNO_OUTPUT_TRAY 0 /* no output tray */
#define OLTPRQXO_TRAY_NR 99 /* output tray max # */
#define OLTPRQXO_TRAY_SORTER 129 /* x'81' otray sorter

/* description of the twoside codes */
/* ENUM _simplex_duplex */
#define OLTPRQXNO_DUPLICATION 0 /* no simplex/duplex */
#define OLTPRQXSIMPLEX 241 /* x'F1' simplex */
#define OLTPRQXDUPLEX 242 /* x'F2' duplex */
#define OLTPRQXTUMBLE_DUPLICATION 243 /* x'F3' tumble duplex

/* description of the sorter codes */
/* ENUM _sorter */
#define OLTPRQXNO_SORTER_GIVEN 0 /* no sorter given */
#define OLTPRQXNO_SORTER 241 /* x'F1' sorter=no */
#define OLTPRQXSORTER_GROUP 242 /* x'F2' sorter=group */
#define OLTPRQXSORTER_COLLATE 243 /* x'F3' sorter=collate */
#define OLTPRQXSORTER_STACKER 244 /* x'F4' sorter=stacker */
#define OLTPRQXSORTER_AUTO 245 /* x'F5' sorter=automat

/* description of the sign codes */
/* ENUM _sign */
#define OLTPRQXPOSITIVE 78 /* positive */
#define OLTPRQXNEGATIVE 96 /* negative

```

```

/* description of extension part */
struct OLTPRQXext_md1 {
#pragma aligned 1
    unsigned char rotation;          /* rotation code */

    /* intray */
    union /* intray_union */ {
        unsigned char intray;      /* input tray number */
        unsigned char intray_format; /* input tray format */
        unsigned char input_tray;  /* input tray code */
    } intray_union;

    /* outtray */
    union /* outtray_union */ {
        unsigned char outtray;     /* output tray number */
        unsigned char output_tray; /* output tray code */
    } outtray_union;
    unsigned char sorter;          /* sorter mode */
    unsigned char left_offset_sign; /* left offset sign */
    unsigned char left_offset;     /* left offset binary;
                                     /* x'00'=no left offset */
    unsigned char top_offset_sign; /* top offset sign */
    unsigned char top_offset;      /* top offset binary; x'00'=
                                     /* no top offset */
    unsigned char duplex;          /* duplex */
    char hdwr_copies[3];           /* hdwr page copies; c'000'
                                     /* or first position=x'00'
                                     /* means no copies */
    char reserved_2[16];          /* free for use */
};

/* parameter area */
struct OLTPRQX_md1 {
#pragma aligned 4

    /* header layout */
    struct ESMFHDR hdr;
    struct OLTPRQXext_md1 extension;
                                     /* extension part layout */
};

```

## C-Spezifikationen

Die C-Schnittstelle implementiert die gleiche Funktionalität wie der OLTPRQX-Assembler-Makro. Die Header-Datei ist in der Include-Datei OLTPRQX.H enthalten. Diese Datei befindet sich in der Standard-RSO-Bibliothek, die mit RSO ausgeliefert wird.

Die Include-Datei ist in die folgenden Komponenten unterteilt:

- Konstanten-Definitionen für spezifische Funktionsgruppen
- Komplexe Datenstrukturen (Datensätze)
- eine Parameterliste mit Referenzen auf die komplexen Datenstrukturen

Die Datei OLTPRQX.H wird in das Anwendungsprogramm mit Hilfe eines Include-Aufrufs in einem C-Anwendungsprogramm eingebunden und die Parameterliste steht zur Verfügung.

Der Standard-Header muss mit Werten initialisiert werden, die mit der Schnittstelle korrespondieren.

Makro-Name	Unit	Funktion	Version
OLTPRQX	275	3	1

```

/* description of the rotation codes */
/* ENUM _rotation */
#define OLTPRQXNO_ROT_GIVEN 0 /* no rotation given */
#define OLTPRQXROT_0 241 /* no rotation */
#define OLTPRQXROT_90 242 /* rotation 90 */
#define OLTPRQXROT_180 243 /* rotation 180 */
#define OLTPRQXROT_270 244 /* rotation 270 */

/* description of the input tray selection mode */
/* ENUM _intray_sel */
#define OLTPRQXBY_FORMAT 255 /* selection via a format */
/* name */
#define OLTPRQXIGNORE 0 /* no input tray given */

/* description of given input tray */
/* ENUM _input_tray */
#define OLTPRQXNO_ITRAY 0 /* no input tray given */
#define OLTPRQXI_TRAY_NR 99 /* max num tray # */
#define OLTPRQXI_TRAY_C5 239 /* x'EF' C5 */
#define OLTPRQXI_TRAY_DL 240 /* x'F0' DL */
#define OLTPRQXI_TRAY_COM10 241 /* x'F1' COM10 */
#define OLTPRQXI_TRAY_MONARCH 242 /* x'F2' monarch */
#define OLTPRQXI_TRAY_DOUBLET 243 /* x'F3' double letter */
#define OLTPRQXI_TRAY_LETTER 244 /* x'F4' letter */
#define OLTPRQXI_TRAY_LEGAL 245 /* x'F5' legal */

```

```

#define OLTPRQXI_TRAY_EXEC 246          /* x'F6' exec          */
#define OLTPRQXI_TRAY_INVOICE 247      /* x'F7' invoice      */
#define OLTPRQXI_TRAY_FOLIO 248       /* x'F8' folio        */
#define OLTPRQXI_TRAY_B5 249          /* x'F9' B5           */
#define OLTPRQXI_TRAY_B4 250          /* x'FA' B4           */
#define OLTPRQXI_TRAY_A5 251          /* x'FB' A5           */
#define OLTPRQXI_TRAY_A4 252          /* x'FC' A4           */
#define OLTPRQXI_TRAY_A3 253          /* x'FD' A3           */
#define OLTPRQXI_TRAY_MANUAL 254      /* x'FE' manual       */

/* description of the output tray code          */
/* ENUM _output_tray                          */
#define OLTPRQXNO_OUTPUT_TRAY 0        /* no output tray     */
#define OLTPRQXO_TRAY_NR 99           /* output tray max #  */
#define OLTPRQXO_TRAY_SORTER 129      /* x'81' otray sorter */

/* description of the twoside codes            */
/* ENUM _simplex_duplex                        */
#define OLTPRQXNO_DUPLEX 0             /* no simplex/duplex  */
#define OLTPRQXSIMPLEX 241            /* x'F1' simplex      */
#define OLTPRQXDUPLEX 242             /* x'F2' duplex       */
#define OLTPRQXTUMBLE_DUPLEX 243      /* x'F3' tumble duplex */

/* description of the sorter codes            */
/* ENUM _sorter                              */
#define OLTPRQXNO_SORTER_GIVEN 0      /* no sorter given    */
#define OLTPRQXNO_SORTER 241          /* x'F1' sorter=no    */
#define OLTPRQXSORTER_GROUP 242       /* x'F2' sorter=group */
#define OLTPRQXSORTER_COLLATE 243     /* x'F3' sorter=collate */
#define OLTPRQXSORTER_STACKER 244     /* x'F4' sorter=stacker */
#define OLTPRQXSORTER_AUTO 245        /* x'F5' sorter=automat */

/* description of the sign codes              */
/* ENUM _sign                                */
#define OLTPRQXPOSITIVE 78             /* positive           */
#define OLTPRQXNEGATIVE 96            /* negative           */

/* description of extension part              */
struct OLTPRQXext_md1 {
    unsigned char rotation;           /* rotation code      */

    /* intray                                  */
    union /* intray_union */ {
        unsigned char intray;         /* input tray number  */
        unsigned char intray_format; /* input tray format  */
        unsigned char input_tray;     /* input tray code    */
    };
};

```

```
    } intray_union;

    /* outtray */
    union /* outtray_union */ {
        unsigned char outtray; /* output tray number */
        unsigned char output_tray; /* output tray code */
    } outtray_union;
    unsigned char sorter; /* sorter mode */
    unsigned char left_offset_sign; /* left offset sign */
    unsigned char left_offset; /* left offset binary;
                                /* x'00'=no left offset */
    unsigned char top_offset_sign; /* top offset sign */
    unsigned char top_offset; /* top offset binary; x'00'=
                                /* no top offset */
    unsigned char duplex; /* duplex */
    char hdwr_copies[3]; /* hdwr page copies; c'000'
                          /* or first position=x'00'
                          /* means no copies */
    char reserved_2[16]; /* free for use */
};

/* parameter area */
struct OLTPRQX_md1 {

    /* header layout */
    struct ESMFHDR hdr;
    struct OLTPRQXext_md1 extension;
};
```

## COBOL-Spezifikationen

Die COBOL-Schnittstelle implementiert die gleiche Funktionalität wie der OLTPRQX-Assembler-Makro.

Die Schnittstelle wird in COBOL mit einer Call-Anweisung und einer Parameter-Area aufgerufen. Die im Folgenden beschriebenen Felder dieser Schnittstelle können mit der Copy-Anweisung in den COBOL-Quelltext innerhalb der Working-Storage-Struktur kopiert werden. OLTPRQX-I-PL erlaubt die Initialisierung der Parameterliste mit den Default-Werten.

OLTPRQXY befindet sich in der Standard-RSO-Bibliothek, die mit RSO ausgeliefert wird.

Die Makros esmhexay und fhdry befinden sich in der Standard-SPOOL-Bibliothek, die mit SPOOL ab V3.0A ausgeliefert wird.

```

000100*****
000200* BEGIN-INTERFACE    OLTPRQX
000300*
000400* TITLE              OLTPRQ extension definition
000500* NAME                OLTPRQXY
000600* DOMAIN              RSO
000700* LANGUAGE            COBOL
000800* COPYRIGHT           (C) Fujitsu Siemens Computers GmbH 2001
000900*                      ALL RIGHTS RESERVED
001000* COMPILATION-SCOPE  USER
001100* INTERFACE-TYPE     CALL
001200* RUN-CONTEXT        TU
001300*
001400* PURPOSE            Interface parameter for OLTPRQ extension
001500*
001600* SYNTAX              Syntax Variant 1:
001700*                      COPY OLTPRQXY
001800*
001900* REMARKS             Welcome to RSO
002000*
002100*****
002200*
002300* VERSION              001
002400* CRDATE              2000-12-14
002500* AUTHOR              F. Langelez
002600* UPDATE              Original issue
002700*
002800* END-INTERFACE      OLTPRQX.
002900*****
003000*****
003100*          OLTPRQXY    - Declarations
003200*****

```

```

003300* parameter area
003400 01 OLTPRQX-PL.
003500* Standard header.
003600 COPY fhdry.
003700 02 OLTPRQX-EXTENSION.
003800* extension part layout
003900 03 OLTPRQX-ROTATION PIC X.
004000* rotation code
004100 88 OLTPRQX-NO-ROT-GIVEN VALUE SYS-X00.
004200* no rotation given
004300 88 OLTPRQX-ROT-0 VALUE "1".
004400* no rotation
004500 88 OLTPRQX-ROT-90 VALUE "2".
004600* rotation 90
004700 88 OLTPRQX-ROT-180 VALUE "3".
004800* rotation 180
004900 88 OLTPRQX-ROT-270 VALUE "4".
005000* rotation 270
005100 03 OLTPRQX-INTRAY PIC X.
005200* input tray number
005300 03 OLTPRQX-INPUT-TRAY REDEFINES OLTPRQX-INTRAY PIC X.
005400* input tray code
005500 88 OLTPRQX-NO-ITRAY VALUE SYS-X00.
005600* no input tray given (0)
005700 88 OLTPRQX-I-TRAY-NR VALUE SYS-X63.
005800* max num tray # (99)
005900 88 OLTPRQX-I-TRAY-C5 VALUE SYS-XEF.
006000* x'EF' C5
006100 88 OLTPRQX-I-TRAY-DL VALUE "0".
006200* x'F0' DL
006300 88 OLTPRQX-I-TRAY-COM10 VALUE "1".
006400* x'F1' COM10
006500 88 OLTPRQX-I-TRAY-MONARCH VALUE "2".
006600* x'F2' monarch
006700 88 OLTPRQX-I-TRAY-DOUBLET VALUE "3".
006800* x'F3' double letter
006900 88 OLTPRQX-I-TRAY-LETTER VALUE "4".
007000* x'F4' letter
007100 88 OLTPRQX-I-TRAY-LEGAL VALUE "5".
007200* x'F5' legal
007300 88 OLTPRQX-I-TRAY-EXEC VALUE "6".
007400* x'F6' exec
007500 88 OLTPRQX-I-TRAY-INVOICE VALUE "7".
007600* x'F7' invoice
007700 88 OLTPRQX-I-TRAY-FOLIO VALUE "8".
007800* x'F8' folio
007900 88 OLTPRQX-I-TRAY-B5 VALUE "9".
008000* x'F9' B5

```



008100	88	OLTPRQX-I-TRAY-B4	VALUE SYS-XFA.
008200*		x'FA' B4	
008300	88	OLTPRQX-I-TRAY-A5	VALUE SYS-XFB.
008400*		x'FB' A5	
008500	88	OLTPRQX-I-TRAY-A4	VALUE SYS-XFC.
008600*		x'FC' A4	
008700	88	OLTPRQX-I-TRAY-A3	VALUE SYS-XFD.
008800*		x'FD' A3	
008900	88	OLTPRQX-I-TRAY-MANUAL	VALUE SYS-XFE.
009000*		x'FE' manual	
009100	03	OLTPRQX-OUTTRAY	PIC X.
009200*		output tray number	
009300	03	OLTPRQX-OUTPUT-TRAY REDEFINES	OLTPRQX-OUTTRAY PIC X.
009400*		output tray code	
009500	88	OLTPRQX-NO-OUTPUT-TRAY	VALUE SYS-X00.
009600*		no output tray given	
009700	88	OLTPRQX-O-TRAY-NR	VALUE SYS-X63.
009800*		output tray max # (99)	
009900	88	OLTPRQX-O-TRAY-SORTER	VALUE SYS-X81.
010000*		x'81' otray sorter	
010100	03	OLTPRQX-SORTER	PIC X.
010200*		sorter mode	
010300	88	OLTPRQX-NO-SORTER-GIVEN	VALUE SYS-X00.
010400*		no sorter given	
010500	88	OLTPRQX-NO-SORTER	VALUE "1".
010600*		x'F1' sorter=no	
010700	88	OLTPRQX-SORTER-GROUP	VALUE "2".
010800*		x'F2' sorter=group	
010900	88	OLTPRQX-SORTER-COLLATE	VALUE "3".
011000*		x'F3' sorter=collate	
011100	88	OLTPRQX-SORTER-SORTER	VALUE "4".
011200*		x'F4' sorter=stacker	
011300	88	OLTPRQX-SORTER-AUTO	VALUE "5".
011400*		x'F5' sorter=automat	
011500	03	OLTPRQX-LEFT-OFFSET-SIGN	PIC X.
011600*		left offset sign	
011700	88	OLTPRQX-LOFPOSITIVE	VALUE SYS-X4E.
011800*		positive (78)	
011900	88	OLTPRQX-LOFNEGATIVE	VALUE SYS-X60.
012000*		negative (96)	
012100	03	OLTPRQX-LEFT-OFFSET	PIC X.
012200*		left offset binary;	
012300*		x'00'=no left offset	
012400	03	OLTPRQX-TOP-OFFSET-SIGN	PIC X.
012500*		top offset sign	
012600	88	OLTPRQX-TOFPOSITIVE	VALUE SYS-X4E.
012700*		positive (78)	
012800	88	OLTPRQX-TOFNEGATIVE	VALUE SYS-X60.

```

012900*          negative (96)
013000          03  OLTPRQX-TOP-OFFSET          PIC X.
013100*          top offset binary;
013200*          x'00'=no top offset
013300          03  OLTPRQX-DUPLEX-MODE        PIC X.
013400*          duplex
013500          88  OLTPRQX-NO-DUPLEX          VALUE SYS-X00.
013600*          no simplex/duplex
013700          88  OLTPRQX-SIMPLEX            VALUE "1".
013800*          x'F1' simplex
013900          88  OLTPRQX-DUPLEX            VALUE "2".
014000*          x'F2' duplex
014100          88  OLTPRQX-TUMBLE-DUPLEX     VALUE "3".
014200*          x'F3' tumble duplex
014300          03  OLTPRQX-HDWR-COPIES        PIC X(3).
014400*          hdwr page copies; c'000' or
014500*          first position=x'00' means no copies
014600          03  OLTPRQX-RESERVED-2        PIC X(16).
014700*          free for use
014800/
014900* parameter area (initialization)
015000 01  OLTPRQX-I-PL.
015100          02  OLTPRQX-I-HDR.
015200*          header layout
015300          03  OLTPRQX-I-IFID.
015400*          header layout
015500          04  OLTPRQX-I-UNIT              PIC S9(4) COMP
015600          VALUE 275.
015700*          unit number
015800          04  OLTPRQX-I-FUNCTION          PIC X
015900          VALUE SYS-X03.
016000*          function
016100          04  OLTPRQX-I-VERSION          PIC X
016200          VALUE SYS-X01.
016300*          version
016400          03  OLTPRQX-I-RC-NBR          PIC S9(9) USAGE BINARY
016500          VALUE -1.
016600*          return code
016700          02  OLTPRQX-I-EXTENSION.
016800*          extension part layout
016900          03  OLTPRQX-I-ROTATION          PIC X
017000          VALUE SYS-X00.
017100*          rotation code
017200          03  OLTPRQX-I-INTRAY           PIC X
017300          VALUE SYS-X00.
017400*          input tray number
017500          03  OLTPRQX-I-OUTTRAY         PIC X
017600          VALUE SYS-X00.

```

017700*		output tray number	
017800	03	OLTPRQX-I-SORTER	PIC X
017900			VALUE SYS-X00.
018000*		sorter mode	
018100	03	OLTPRQX-I-LEFT-OFFSET-SIGN	PIC X
018200			VALUE SYS-X4E.
018300*		left offset sign	
018400	03	OLTPRQX-I-LEFT-OFFSET	PIC X
018500			VALUE SYS-X00.
018600*		left offset binary;x'00'=no left offset	
018700	03	OLTPRQX-I-TOP-OFFSET-SIGN	PIC X
018800			VALUE SYS-X4E.
018900*		top offset sign	
019000	03	OLTPRQX-I-TOP-OFFSET	PIC X
019100			VALUE SYS-X00.
019200*		top offset binary; x'00'=no top offset	
019300	03	OLTPRQX-I-DUPLEX	PIC X
019400			VALUE SYS-X00.
019500*		duplex	
019600	03	OLTPRQX-I-HDWR-COPIES	PIC 9(3)
019700			VALUE 000.
019800*		hdwr page copies; c'000' or	
019900*		first position=x'00' means no copies	
020000	03	OLTPRQX-I-RESERVED-2	PIC X(16)
020100			VALUE LOW-VALUES.

**Beispiel**

```

IDENTIFICATION DIVISION.
PROGRAM-ID. tutmexcb.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SPECIAL-NAMES.
    TERMINAL IS v-terminal,
    SYMBOLIC CHARACTERS
    COPY esmhexas.
DATA DIVISION.
WORKING-STORAGE SECTION.
COPY oltpqrxy.
PROCEDURE DIVISION.
MAIN SECTION.
p-main.
    PERFORM s-test001.
    PERFORM s-test002.
p-exit.
    STOP RUN.
*
* TEST 001 : use default values

```

```
*
s-test001 SECTION.
*-----*
*
p-test001-strt.
*-----*
*
      DISPLAY "TEST1 (DFT VALUES) OF TUTMEXTCB" UPON v-terminal.
      MOVE oltpqrq-i-p1 TO oltpqrq-p1.
      DISPLAY oltpqrq-p1 UPON v-terminal.
*
p-test001-exit.
*-----*
*
      EXIT.
*
* TEST 002 : initialise some fields
*
s-test002 SECTION.
*-----*
*
p-test002-strt.
*-----*
*
      MOVE oltpqrq-i-p1 TO oltpqrq-p1.
*
* select rotation=0, input tray= 3, output tray=sorter stacker,
* left offset of +4, top offset of -4,no duplex selection and one
* hardware copy.
*
      SET oltpqrq-rot-0 TO TRUE.
      MOVE SYS-X03 TO oltpqrq-input-tray.
      SET oltpqrq-o-tray-sorter TO TRUE.
      SET oltpqrq-sorter-stacker TO TRUE.
      SET oltpqrq-lofpositive TO TRUE.
      MOVE SYS-X04 TO oltpqrq-left-offset.
      SET oltpqrq-tofnegative TO TRUE.
      MOVE SYS-X04 TO oltpqrq-top-offset.
      SET oltpqrq-no-duplex TO TRUE.
      MOVE "001" TO oltpqrq-hdwr-copies.
*
p-test002-exit.
*-----*
*
      EXIT.
```

## PRNTDOC (für SPOOL, SPS, RSO und Dprint) - Datei ausgeben

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprachen:** Assembler, C, COBOL

**Makrotyp:** S

Mit dem Makro PRNTDOC können Sie Dateien auf Drucker ausgeben lassen. Durch Angabe entsprechender Operanden können Sie den Druckauftrag steuern. Enthält ein Druckauftrag außer dem Namen der zu druckenden Datei keine weiteren Angaben, dann setzt SPOOL zur Erledigung des Auftrags Standardwerte ein.

Die Angaben aus dem Kommando PRNTDOC werden zusammen mit den Festlegungen (Voreinstellungen) zur Geräteauswahl in einen SPOOL-Control-Block (SCB) eingetragen. Der SCB wird in eine der SPOOLOUT-Warteschlange (lokaler SPOOLOUT, RSO) eingehängt und als eigener Auftrag mit eigener TSN verwaltet.

Sie können mit dem Kommando SHOW-PRINT-JOB-STATUS die Verarbeitung Ihres SPOOLOUT-Auftrages verfolgen, bestimmte Parameter eines Druckauftrags mit dem Kommando MODIFY-PRINT-JOB-ATTRIBUTES korrigieren, mit dem Kommando CHANGE-TASK-PRIORITY die Priorität verändern und mit dem Kommando CANCEL-PRINT-JOB den SPOOLOUT-Auftrag abbrechen.



Der Makro PRNTDOC stellt eine Erweiterung der Funktionalität des Makros PRNT dar. Dessen Funktionsumfang entspricht der SPOOL-Version 2.7B, d.h. die Erweiterungen und Ergänzungen der neuen SPOOL-Versionen ab 3.0A können nur mit PRNTDOC genutzt werden.

Ab SPOOL V3.0 wird die Funktionalität dieses Makros nicht nur (wie bisher im PRNT) für Programmierer in Assembler, sondern zusätzlich in den Programmiersprachen C und COBOL angeboten. Für Assembler ist ein „konventionelles“ Aufruf-Format verfügbar, C-Programmierer können die PRNTDOC-Funktionen über einen SVC-Aufruf nutzen und COBOL-Programmierer können auf ein spezielles Modul aus der Laufzeit-Bibliothek zugreifen, das die PRNTDOC-Funktionen realisiert. Die für die Nutzung des Makros benötigten Datenbeschreibungen und jeweils ein Beispiel in den verschiedenen Programmiersprachen finden Sie im Anschluss an die Operandenbeschreibung ab [Seite 119](#).

## SPOOLOUT steuern

### Drucker auswählen

Enthält ein Druckauftrag keine Angaben über den zu verwendenden Drucker, dann vergibt SPOOL den Auftrag an einen beliebigen verfügbaren Schnelldrucker. Die Zahl der Zeichen pro Zeile, die dieser druckt, hängt vom Formular ab.

Soll der Auftrag auf einem bestimmten Drucker ausgegeben werden, dann geben Sie bitte innerhalb des Operanden PRTYPE die entsprechende Druckerbezeichnung an (siehe Operandenbeschreibung auf [Seite 96](#)).

Die nachfolgend aufgeführten Operanden stellen Mindestanforderungen bezüglich des Druckers dar. Geben Sie einen der aufgeführten Operanden an, dann nehmen Sie damit gleichzeitig Einfluss auf die Druckerauswahl.

Operand	notwendiger Drucker (Mindestanforderung)
PRNAME=gerät	RSO-Drucker
PRTYPE=*LP65_PRINTER	LP65-Drucker
PRTYPE=*APA_PRINTER	APA-Drucker
PRTYPE=*HP_PRINTER	HP-Drucker oder HP90-Drucker
USERRES [≡*DUMMY]	HP-, HP90- oder APA-Drucker
PAGECOP [≡*STD und ≡0]	HP-, HP90- oder LP65-Drucker
PAGEDEF=<name>	APA-Drucker
POOLNAM	HP-Drucker oder HP90-Drucker
FOB	HP-Drucker oder HP90-Drucker
ROT	HP-Drucker oder HP90-Drucker
OVERLAY (<face>,<reverse>)	LP65-Drucker

Wird keiner der Operanden in dieser Tabelle angegeben, so gilt der Standardwert für die Druckerwahl aus der SPOOL-Parameterdatei.

## Formular angeben

Enthält ein Druckauftrag keine Angaben über das zu verwendende Formular, dann setzt SPOOL das Standardformular "STD" ein, das in der SPOOL-Parameterdatei definiert ist.

Standardmäßig verweist das Standardformular "STD"

- auf den Standard-Loop "C1" mit einer Zeilendichte von 6 lpi und dem Vertikaltabulator "Kanal 1" (standardmäßig Seitenbeginn) auf Zeile 3;
- auf den Standard-Rotation-Loop R06 für HP-Drucker und HP90-Drucker mit einer Zeilendichte von 6 lpi und einer Formularlänge von 13,5 Zoll.

Soll ein anderes Formular verwendet werden, geben Sie das gewünschte Formular bitte mit FORM=... an.

## Papiervorschub steuern

Zu jedem Formular gibt es die entsprechenden Papiervorschubinformationen (Loop). Wenn Sie keine anderen Angaben machen, wird automatisch der Loop verwendet, der dem verwendeten Formular zugeordnet ist. Mit LOOP=... können Sie aber auch einen anderen Loop angeben. Für alle Drucker muss dieser Loop in der System-PRFILE \$SYSSPOOL.PRFILE oder einer entsprechenden Benutzer-PRFILE enthalten sein.



Bei expliziter Angabe eines Loops müssen Sie darauf achten, dass die Loop-Länge der Formularlänge entspricht.

Bei Druckern mit ladbarem Vorschubinformationspuffer wird durch die Angabe eines Papierformulars (Operand FORM) der zu diesem Formular gehörige Loop in den Puffer geladen. Nur wenn in demselben SPOOLOUT-Auftrag zugleich der Operand LOOP angegeben ist, wird der Vorschub mit diesem bei LOOP angegebenen Loop geregelt. In jedem Fall muss der aufgerufene Loop in einer PRFILE vorhanden sein.

Bei fehlenden Angaben zu Papierformular und Font setzt SPOOL voraus, dass im Loop der Vertikaltabulator "Kanal 1" auf den Beginn einer neuen Seite (Zeile 3) gesetzt ist.

Der Standard-Loop mit dem Namen "C1", der Standard-Font "101" und weitere für Laserdrucker bestimmte Fonts sind standardmäßig in der System-PRFILE enthalten.

## Loops selbst definieren

Eigene Loops können Sie mit dem Programm PRM definieren und verwalten. Näheres siehe Handbuch „[PRM \(BS2000/OSD\)](#)“.

### Vorschubsteuerzeichen

EBCDIC-Vorschubsteuerzeichen:

Vorschubsteuerzeichen	Wirkung
X'40' bis X'4F'	n Zeilenvorschübe vor dem Drucken, 1 Zeilenvorschub nach dem Drucken
X'00' bis X'0F'	n Zeilenvorschübe nach dem Drucken
X'C1' bis X'CB'	Kanalvorschub vor dem Drucken
X'81' bis X'8B'	Kanalvorschub nach dem Drucken

Aufbau des Druckersteuerbytes von SIEMENS (EBCDIC) und IBM:

Wert	Bit-Position und Bedeutung (EBCDIC)							
	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
0	Zeilen- vor-schub	nach dem Drucken	immer 0	immer 0	Anzahl der Zeilen bzw. V- Tab-Nr. Wert: 8	Anzahl der Zeilen bzw. V- Tab-Nr. Wert: 4	Anzahl der Zeilen bzw. V- Tab-Nr. Wert: 2	Anzahl- der Zeilen bzw. V- Tab-Nr. Wert: 1
1	Kanal- vor-schub	vor dem Drucken						

Wert	Bit-Position und Bedeutung (IBM)							
	2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
0	Zeilen- vor-schub	V-Tab- nummer	V-Tab- nummer	Anzahl Zeilen bzw. V- Tab-Nr. Wertig- keit 2	Anzahl Zeilen bzw. V- Tab-Nr. Wertig- keit 1	immer 0	nach dem Drucken	immer 1
1	Kanal- vor-schub	Wertig- keit 8	Wertig- keit 4	Wertig- keit 2	Wertig- keit 1		vor dem Drucken	

\*) V-Tab ist der Vertikaltabulator "Kanal n"



ASA-Vorschubsteuerzeichen und entsprechende EBCDIC-Vorschubsteuerzeichen:

ASA-Vorschub	EBCDIC-Vorschub	Wirkung
C'+'	X'00'	kein Zeilenvorschub nach dem Drucken
C'-'	X'42'	2 Zeilen Vorschub vor dem Drucken
C'0'	X'40'	1 Zeile Vorschub vor dem Drucken
C'1'	X'C1'	Vorschub auf "Kanal 1" vor dem Drucken
.	.	.
.	.	.
.	.	.
C'A'	X'CA'	Vorschub auf "Kanal 10" vor dem Drucken
C'B'	X'CB'	Vorschub auf "Kanal 11" vor dem Drucken
andere	X'40'	Kein Zeilenvorschub vor dem Drucken

Bei Laserdruckern werden unzulässige Steuerzeichen berichtigt.



Der Vertikaltabulator "Kanal 12" (X'CC') oder (X'8C') ist für SPOOL reserviert.

Auf Grund von Unterschieden in der Druckerhardware entsprechen sich ASA- und Siemens-Vorschubsteuerzeichen nicht völlig:

Siemens-Drucker schieben nicht vor, sondern stets **nach** dem Drucken eine Zeile vor, außer bei X'00' (siehe oben).

### Fonts angeben

Jedem Formular, das in der SPOOL-Parameterdatei eingetragen ist, ist ein Font zugeordnet. Diese Zuordnung ist in der SPOOL-Parameterdatei eingetragen.

Wird im PRNTDOC kein Font angegeben, dann wird standardmäßig der Font „101“ verwendet. Dieser Font ist dem Formular "STD" in der SPOOL-Parameterdatei zugeordnet, und zwar für alle Druckertypen bis auf die Druckertypen 3337, 3338 und 3339. Der Font "101" hat die Schriftart OCR-B und ist definiert für eine Zeilendichte von 6 Zeilen/Zoll (6 lpi) und eine Zeichendichte von 10 Zeichen/Zoll (10 cpi).

Mit dem Operanden CHARSET können Sie bei HP/HP90-Drucker einen Font aus der System-PRFILE direkt angeben. Steht der gewünschte Font in einer Benutzer-PRFILE, muss diese gleichzeitig mit dem Operanden USERRES angegeben werden.

Beim Operanden CHARSET geben Sie

- entweder die Namen von maximal vier Fonts (für HP-/HP90- Drucker)
- oder den Namen eines Font-Pools mit maximal 64 Fonts (HP-/HP90-Drucker)

an, mit denen eine Datei ausgedruckt werden soll. Die Fonts sind in der Ressourcen-Bibliothek \$SYSSPOOL.PRFILE enthalten. Sie können sich aber auch ein private Benutzer-PRFILE erstellen, die einen oder mehrere eigene, mit dem Programm PRM erstellte Fonts enthält.

In einer Benutzer-PRFILE können Sie sich auch einen Font-Pool selbst definieren. Soll eine Datei mit einem eigenen Font ausgedruckt werden, müssen Sie mit dem Operanden USERRES den Dateinamen angeben.

Auch bei Drucken im APA-Modus können Sie direkt einen Font aus der Standard-SPSLIB angeben. Steht der gewünschte Font in einer Benutzer-SPSLIB, muss diese analog zu HP-/HP90-Druckern gleichzeitig mit dem Operanden USERRES angegeben werden. Die Fonts sind in der Drucksteuerdatei \$SYSSPOOL.SYSPRT.SPS.021 enthalten. In einer Benutzer-SPSLIB können Sie mit dem Softwareprodukt SPM eingeschränkt Fonts selberstellen. Genauer dazu siehe im Handbuch „[PRISMAproduction/BS2000 - SPS](#)“. Soll eine Datei mit einem eigenen Font ausgedruckt werden, müssen Sie mit dem Operanden USERRES den Dateinamen angeben.

Vor Beginn des Dateiausdrucks lädt das SPOOL-System die angegebenen Fonts aus der angegebenen PRFILE in den Zeichenspeicher des Laserdruckers. Wird eine Datei mit DOCFORM=\*TEXT im Makro PRNTDOC ausgedruckt, d.h. keine Steuerzeichenauswertung im Text (Ausnahme: Druckertyp 3365, siehe Handbuch „[SPOOL \(BS2000/OSD\)](#)“), wird für den Ausdruck der gesamten Datei der erste im Operanden CHARSET des Makros PRNTDOC angegebene Font verwendet. Es ist also sinnvollerweise nur ein Font anzugeben. Sind dagegen Steuerzeichen im Text der Druckdatei enthalten, die von SPOOL interpretiert werden sollen – d.h. im PRNTDOC muss CONT-MOD=\*PAGE\_MODE angegeben sein - dann wird der jeweils zu verwendende Font aus den geladenen Fonts auf folgende Art bestimmt:

- Die Steuerinformation im ersten Datensatz jeder Druckseite legt u.a. fest, mit welchem Font der Druck der Seite begonnen wird. Durch Steuerzeichen innerhalb der Druckdaten kann an jeder beliebigen Stelle ein Wechsel des Fonts erfolgen, d.h. im Extremfall bei jedem Zeichen.
- Die bei CHARSET angegebenen Fonts werden bei der Bearbeitung des Kommandos gemäß dieser Angabe in den Zeichenspeicher des Druckers geladen.
- Soll von einem Font auf einen anderen mittels Steuerzeichen umgeschaltet werden, so geschieht dies folgendermaßen:

Alle Fonts, die Sie angegeben haben (explizit oder durch Angabe eines Pool-Namens), werden zum Zeitpunkt des Scheduling auf dem gewünschten Gerät geladen.

Bei DOCFORM=\*TEXT kann nur der erste angegebene Font benutzt werden. Die Anzahl der Fonts (explizit angegeben im PRNTDOC oder im angegebenen Pool) werden verglichen mit der Obergrenze, die als globaler Parameter bei der Installation angegeben wurde.

Der Scheduler prüft, ob das angegebene Gerät genügend Speicherplatz besitzt, um alle (explizit oder implizit angegebenen) Fonts laden zu können.

Sie sollten sich möglichst auf die von der Systembetreuung eingerichteten Pools beziehen.

- entweder durch Angabe des Pool-Namens allein
- oder durch Angabe eines Pool-Namens und eines Index

Der Scheduler bildet Gruppen von Aufträgen, die denselben Font bzw. denselben Pool benutzen und verhindert so ein unnötiges Umladen von Fonts.

Werden im PRNTDOC einzelne Fonts angegeben, die noch nicht im Zeichenspeicher des Geräts geladen sind und sind dort auch keine freien Abschnitte mehr vorhanden, dann sucht der Druckercontroller unter den geladenen Fonts diejenigen aus, die am seltensten aufgerufen wurden und tauscht sie gegen die neu angeforderten aus.

Werden mehr als vier Fonts angefordert, und ist keiner der zugreifbaren HP-Laserdrucker in der Lage, die geforderte Anzahl von Fonts zu laden, dann kann der Auftrag nicht verarbeitet werden.

In den Informationen, die Sie mit dem Kommando SHOW-USER-STATUS anfordern, sind im Feld OPT alle Aufträge mit "\*" gekennzeichnet, die mehr als vier Fonts ansprechen oder Dias oder das Seitendrehmodul. Die Anzahl der angeforderten Fonts wird angefordert mit dem Kommando SHOW-PRINT-JOB-STATUS.

Beim Absetzen des Makros PRNTDOC mit den Operanden LOOP, CHARSET usw. für Laserdrucker wird in der Ressourcen-Bibliothek \$SYSSPOOL.PRFILE geprüft, ob die entsprechenden Einträge vorhanden sind. Falls nicht, wird der Makro PRNTDOC abgewiesen.

Beim Absetzen des Makros PRNTDOC mit dem Operanden FORM und/oder LOOP für Drucker mit ladbarem VFB wird in der Ressourcen-Bibliothek geprüft, ob der entsprechende Eintrag vorhanden ist. Falls nicht, wird der Makro PRNTDOC abgewiesen.

### *Beispiel*

Eine Datei soll mit 4 verschiedenen Fonts auf Laserdrucker ausgegeben werden:

```
PRNTDOC , . . . , CHARSET=(100,360,370,#XY) , . . .
```

Soll z.B. auf den Font 370 (3. Font) umgeschaltet werden, so ist entsprechend der gewünschten Zeichendichte eines der folgenden Steuerzeichen auszuwählen:

X'06'	Zeichendichte 10 Zeichen/Zoll und Font im MXM-Abschnitt 3
X'0A'	Zeichendichte 12 Zeichen/Zoll und Font im MXM-Abschnitt 3
X'0E'	Zeichendichte 15 Zeichen/Zoll und Font im MXM-Abschnitt 3

Dabei ist natürlich zu berücksichtigen, für welche Zeichendichte der Font 3 (370) ursprünglich definiert wurde.

### Layout-Steuerzeichen auswerten

Wenn eine Datei ausgedruckt werden soll, die Layout-Steuerzeichen enthält, können Sie bei Auftragserteilung festlegen, ob diese Steuerzeichen ausgewertet werden sollen oder nicht, und zwar durch Angabe des Operanden DOCFORM mit entsprechendem Operandenwert. Zu den Steuerzeichen siehe Handbuch „[SPOOL \(BS2000/OSD\)](#)“.

Das notwendige Laden der Font- und Vorschubpuffer übernimmt SPOOL bei der Bearbeitung des SPOOLOUTs.



Der Operator kann mit dem Kommando START-PRINTER-OUTPUT festlegen, dass Laserdruckeraufträge, die mit der Angabe DOCFORM=\*TEXT gegeben wurden, auch auf einem Anschlagdrucker (Zeilendrucker) ausgegeben werden dürfen. Die Operanden LOOP, PAGECOP, ADDCOP, USERRES, CHARSET und OVERLAY werden in diesem Fall nicht ausgewertet.

### Beispiele für Steuerzeichen in einer Datei

Druckzeichen
--------------

Der Datensatz besteht nur aus abdruckbaren Zeichen (Ohne Vorschubsteuerzeichen)

V	IK	Druckzeichen
---	----	--------------

Der Datensatz beginnt mit einem Vorschubsteuerzeichen (V), gefolgt von Text (einschließlich ISAM-KEY (IK) bei ISAM-Dateien).

IK	V	Druckzeichen
----	---	--------------

Der Datensatz beginnt mit dem ISAM-Schlüssel (KEY-POS=5), danach folgt das Vorschubsteuerzeichen (V) und der Text.

Druck- und Steuerzeichen
--------------------------

Der Datensatz enthält Text gemischt mit Steuerzeichen (z.B. Font wechseln).

## Formulare, Loops und Fonts selbst definieren

Bei HP/HP90-Druckern:

Papierformulare, Loops und Fonts werden in der Regel von der Systembetreuung definiert. Sie können jedoch auch selbst Loops und Fonts erstellen, in einer benutzereigenen Ressourcen-Bibliothek (PRFILE) ablegen und für SPOOL-Aufträge verwenden. Die Loops und Fonts müssen sich auf ein Papierformular beziehen, das in der SPOOL-Parameterdatei bereits vorhanden ist. Durch Angabe des Operanden USERRES veranlassen Sie SPOOL, Ihre private Ressourcen-Bibliothek für diesen Auftrag heranzuziehen. Näheres darüber finden Sie im Handbuch „[PRM \(BS2000/OSD\)](#)“.

Bei Druckern im APA-Modus:

Sie können auch bei Druckern im APA-Modus eine private Drucksteuerdatei (SPSLIB) mit eigenen Fonts usw. erzeugen. Auch das geben Sie mit dem Operanden USERRES an. Siehe dazu auch Handbuch „[PRISMAproduction/BS2000 - SPS](#)“.

## SPOOLOUT mit Jobvariable überwachen

SPOOL-Aufträge lassen sich mit Jobvariablen überwachen. Geben Sie im Makro PRNTDOC den Namen eine Monitorvariablen (MONJV) an, so legt SPOOL dort laufend Informationen über die Verarbeitung des Auftrags ab.

Ab BS2000 V10.0 unterliegen diese SPOOL-Jobvariablen nicht der Kontrolle des Softwareprodukts "JV", sondern werden von SPOOL selbst kontrolliert. Das bedeutet, dass sie auch von SPOOL gegen Zugriffe anderer Aufträge geschützt wird. Ist eine Jobvariable beispielsweise einem SPOOLOUT-Auftrag zugeordnet, kann sie nicht für einen anderen Auftrag angefordert oder benutzt werden. Das ist unabhängig davon, ob der SPOOLOUT-Auftrag in der Warteschlange steht, zurückgestellt wurde oder gerade bearbeitet wird.

SPOOL-Jobvariable können auch mit einem Kennwort geschützt werden. Soll eine kennwortgeschützte SPOOL-Jobvariable abgefragt werden, muss zuvor das Kennwort mit dem Kommando ADD-PASSWORD in die Kennworttabelle eingetragen werden oder mit dem Operanden JVPASSW im Makro PRNTDOC angegeben werden.

## Fehlerverarbeitung bei SPOOL-Jobvariablen

Gibt es Zugriffprobleme bei SPOOL-Jobvariablen, kann SPOOL folgendermaßen reagieren:

- Eignet sich der Fehler bei der Syntaxprüfung des Makros PRNTDOC, wird der Makro zurückgewiesen, und zwar für den Operanden, bei dem sich der Fehler ereignet. Bei gemeinsamer Bearbeitung mehrerer Aufträge (Operand FAMILY) wird nur die entsprechende Datei zurückgewiesen. Die Kommandoverarbeitung wird anschließend fortgesetzt. Ist nur ein einziger Auftrag in Bearbeitung, wird dieser abgebrochen.

Wird im Makro PRNTDOC der Operand STPROC=\*AT\_FILE\_CLOSING angegeben, dann setzt SPOOL die Verarbeitung trotz des Problems mit der Variablen fort, da andernfalls ein Teil der Daten gedruckt werden würde. Stattdessen gibt SPOOL eine Warnung an den Benutzer aus und setzt die Auftragsverarbeitung ohne Monitorvariable fort.

- Tritt der Fehler während des SPOOLOUTs auf, gibt SPOOL die Meldung SPS0450 auf dem Schlussblatt aus, setzt aber die SPOOLOUT-Verarbeitung ansonsten normal fort.

## Inhalt der SPOOL-Jobvariable

Die Jobvariable ist folgendermaßen strukturiert:

Sta	TSN	Proc.	Reserve	S	OTSN	E#	EC#	Device	ERCOD		ERMSG
0	3	8	16	37	39	44	48	52	61		70

Die ersten drei Felder der Jobvariablen sind ENTER-Jobs vorbehalten. Die nachfolgenden Felder sind SPOOL zugeordnet und spiegeln die verschiedenen Prozesse bei der SPOOL-Job-Verarbeitung wider.

Die Felder "Status" und "Processor" werden linksbündig besetzt und sind zunächst mit Leerzeichen gefüllt, die übrigen Felder werden rechtsbündig besetzt und sind zunächst ebenfalls mit Leerzeichen gefüllt. Existiert die Jobvariable nicht, wird sie bei der Gültigkeitsprüfung des Kommandos gegebenenfalls erzeugt. Der Inhalt einer Jobvariablen kann mit dem Kommando SHOW-JV abgefragt werden.

## Übersicht

Feld	Inhalt	Bedeutung
Sta	\$S \$R \$P  \$T \$A \$K	(spooled-in) Auftrag angenommen und in Wartestatus (running) Auftrag aktiv (pre-processing) Auftrag, bei dem ein Abschnitt gegenwärtig vorverarbeitet wird (terminated) Auftrag erfolgreich beendet (aborted) Auftrag gestrichen oder abnormal beendet (kept) Auftrag wurde zurückgehalten
TSN	tsn	enthält die TSN des Auftrags mit vorangestelltem Füllzeichen „0“; Im Falle eines Replay-Auftrags ist tsn die aktuelle TSN. Die ursprüngliche TSN befindet sich in einem anderen Feld;
Proc.	proc	enthält wie bei ENTER-JOB-Aufträgen die Katalogkennung des Systems (Home-Pubset), auf dem der Auftrag gegeben wurde oder das Band eingespielt wird;
Res.		reserviert
S	0 1 2 3	gibt an, ob der Auftrag mit Bandverarbeitung zu tun hat Auftrag enthält nicht vorverarbeitete Abschnitte allgemeine Auftragsverarbeitung Auftrag ist auf Band oder wurde dort abgelegt Auftrag kommt von einem Band
OTSN	otsn	enthält im Falle eines Replay-Auftrags die ursprüngliche TSN;
E#	nnn	Anzahl der Exemplare, die gedruckt werden sollen
EC#	nnn	Anzahl der bereits verarbeiteten Exemplare
Dev.	xxxxxx	gibt für aktive Aufträge (\$R, \$T oder \$A) den Gerätenamen, die Gerätekurzbezeichnung oder "TP" für Benutzerbänder an;
ERCOD	xxxxxxx	Fehlercode für RSO-Aufträge
ERMSG	xxxxxxx	Identifikationsnummer der Fehlermeldung bei RSO-Aufträgen



Normalerweise hat ein Auftrag nacheinander den Status \$S, \$R und \$T (oder \$A / \$K). Ein Replay-Auftrag erhält den Status \$T, nachdem die Datei zum Band übertragen wurde. Dieses Band kann auf demselben System abgespielt werden. In diesem Fall wird die Jobvariable geprüft, sobald die Datei im System ist. Existiert die ursprüngliche Jobvariable dann noch, dann wird sie neu initialisiert und erhält den Status \$S.

Das Jobvariablen-Subsystem ist optional. Es kann während des Systemlaufs jedoch nicht entladen werden. Wenn es während der Gültigkeitsprüfung des Makros PRNTDOC nicht verfügbar ist und eine Jobvariable wird angefordert, dann weist

SPOOL das Makro zurück. Die SPOOL-Jobvariablen können nicht in den Kommandos SHOW-PRINT-JOB-STATUS, CANCEL-PRINT-JOB oder CHANGE-TASK-PRIORITY angegeben werden.



**Kurzbeschreibung der PRNTDOC-Funktionen**

<b>Operand</b>	<b>Funktion</b>
ADDCOP	Fordert zusätzliche Ausdrücke der Datei an
CALLER	Vereinbart den Aufrufer des Makros
CHARSET	Wählt Fonts für die Ausgabe aus
CHECKP	Legt den Modus für die Checkpoint-Verarbeitung fest
CONTMOD	Vereinbart die Art der Steuerzeichen-Interpretation
COVPAGE	Legt Angaben für Exit-Routinen für das Drucken von Deck- und Schlussblättern fest
DELF	Legt fest, ob die Datei nach Beendigung der Ausgabe gelöscht werden soll
DNAME	Vereinbart den Namen der auszudruckenden katalogisierten Datei und die Länge des Dateinamens
DOCFORM	Legt fest, welcher Art die Dokumentinhalte sind
DPPAR	Gibt die Adresse einer Parameterliste an, in der Distributed Print Services-spezifische Werte für den Aufruf des Makros hinterlegt sind
DSEMPAR	Gibt die Adresse einer Parameterliste an, in der DSEM-spezifische Werte für den Aufruf des Makros hinterlegt sind
DTYPE	Vereinbart den Typ der auszugebenden Dateien
EAMN	Vereinbart die Nummern der auszugebenden EAM-Bindemoduldateien
EFO	Legt fest, ob ein Filmdia - für HP90- und 3365 Drucker ein EFO-Datendia - zu verwenden ist
FAMILY	Legt fest, ob bei Angabe von mehreren Dateien oder Bibliothekselementen in einem Makro PRNTDOC eine gemeinsame TSN vergeben wird
FIRSREC	Vereinbart, ob von der Eingabedatei nur eine bestimmte Anzahl von Datensätzen verarbeitet werden soll
FOB	Legt fest, ob ein FOB-Datendia für die Verarbeitung des SPOOLOUT-Auftrags zu verwenden ist
FORM	Bezeichnet das Papierformat, das für die Ausgabe verwendet werden soll
FRMNAME	Art des Dokumenteninhalts
HEADLIN	Legt fest, ob auf jede Seite (ausgenommen Deck- und Schlussblatt) eine Überschriftzeile zu drucken ist
INTRAY	Legt das Papiereingabefach für den Druckertyp LP65 fest
INTRAYF	Legt das Papiereingabefach über ein Schlüsselwort fest
JVPASSW	Gibt das Kennwort an, mit dem die Jobvariable geschützt worden ist
LASTREC	Vereinbart, ob von der Eingabedatei nur eine bestimmte Anzahl von Datensätzen verarbeitet werden soll

(Teil 1 von 3)

<b>Operand</b>	<b>Funktion</b>
LEFTMAR	Legt fest, ob der Ausgabertext um die angegebene Anzahl Spalten eingerückt werden soll
LEFTOFF	Legt den Abstand des ersten zu druckenden Zeichens vom linken Rand der physikalischen Seite in Millimetern fest.
LIBELEM	Vereinbart den Namen, Version und Typ eines auszudruckenden Elements aus einer PLAM-Bibliothek
LINEPP	Legt fest, wieviel Zeilen auf eine Seite zu drucken sind
LINESP	Bestimmt die Anzahl der Zeilenvorschübe bzw. die Art der Steuerzeichenauswertung
LOCKF	Legt fest, ob die Datei geschützt werden soll, solange sich der SPOOLOUT-Auftrag im Wartezustand befindet
LOOP	Vereinbart einen oder mehrere Loops
LSTNN	Vereinbart Namen und Nummer einer SYSLST- bzw. SYSOUT-Systemdatei
MACID	Bestimmt das zweite bis einschließlich vierte Zeichen der Feldnamen und Equates
MF	Typ des Makroaufrufs
MONJV	Gibt die Jobvariable an, in der Informationen über die Auftragsverarbeitung abgelegt werden sollen
OUTFORM	Gibt an, ob nur im Zeichenformat oder zusätzlich sedezimal auszugeben ist
OUTPART	Vereinbart, ob sich die Ausgabe auf eine Untermenge aller logischen Druckseiten beschränken soll
OUTTRAY	Legt das Papierausgabefach für den Druckertyp LP65 fest
OVERLAY	Vereinbarung von EFO-Datendias für Drucker vom Typ LP65
PAGECOP	Legt die Anzahl der Seitenkopien fest
PAGEPCL	Gibt an, welche Seiten-Definition für den Ausdruck auf LP65- oder APA-Druckern benutzt werden soll
PARAM	Bezeichnet die Adresse der Operandenliste
POOLIND	Vereinbart die Nummer des Fonts aus einem Font-Pool
POOLNAM	Vereinbart den Namen des Font-Pools
PREFIX	Bestimmt das erste Zeichen der Feldnamen und Equates
PRJCLAS	Bestimmt die Auftragsklasse des SPOOLOUT-Auftrags
PRJNAME	Legt den Auftragsname für den SPOOLOUT-Auftrag fest
PRJPRIO	Bestimmt die Dringlichkeit des SPOOLOUT-Auftrags
PRNAME	Spezifiziert den angeforderten Zieldrucker oder Drucker-Pool des Druckauftrags

(Teil 2 von 3)

<b>Operand</b>	<b>Funktion</b>
PROCADM	Legt fest, ob der SPOOLOUT-Auftrag unter der eigenen oder einer anderen angegebenen Benutzerkennung und Abrechnungsnummer ausgeführt werden soll
PROGRAM	Gibt den Gerätenamen eines virtuellen Druckers an
PRTYPE	Gibt an, welcher Druckertyp den Druckauftrag verarbeiten soll
RECPART	Vereinbart, dass von allen ausgewählten Datensätzen nur ein bestimmter Teil jedes Satzes verarbeitet werden soll
ROT	Legt fest, ob die zu druckenden Seiten des SPOOLOUT-Auftrags gedreht auszugeben sind
ROTLOOP	Bestimmt den Loop, mit dem die Ausgabe im Querformat gesteuert werden soll
RSOPAR	Gibt die Adresse einer Parameterliste an, in der RSO-spezifische Werte für den Aufruf des Makros hinterlegt sind
SECTREC	Vereinbart, ob die Datei mit Hilfe von Abschnittsmarken strukturiert werden soll
SCHEDTIME	Gibt die Zeit an, ab der der Druckauftrag abgearbeitet bzw. gedruckt werden kann. Ab diesem Zeitpunkt unterliegt der Auftrag den Standardregeln zur Abarbeitung eines Druckauftrags
SPSPAR	Gibt die Adresse einer Parameterliste an, in der SPS-spezifische Werte für den Aufruf des Makros hinterlegt sind
SRTMODE	Legt den Sortiermechanismus für das Papierausgabefach fest
STPROC	Vereinbart den Zeitpunkt des Ausdrucks der Systemdatei vor Beendigung des Auftrags
TRANTAB	Legt fest, ob zur Verarbeitung des SPOOLOUT-Auftrags eine Code-Umsetzungstabelle zu verwenden ist
TRUNC	Legt das Verhalten für den Fall fest, dass Zeilen abgeschnitten werden
TWOSIDE	Legt fest, ob auf dem LP65-Drucker einseitig oder doppelseitig gedruckt werden soll
TOPOFF	Legt den Abstand der ersten zu druckenden Zeile vom oberen Rand der physikalischen Seite in Millimetern fest.
USERRES	Bezeichnet eine Benutzerdatei, die alle benötigten Ressourcen für den Ausdruck auf unterschiedlichen Druckertypen enthält
VARIANT	Variante der generierten Parameterliste
VIRTUAL	Legt fest, ob ein Auftrag über einen virtuellen Drucker weitergegeben wird

(Teil 3 von 3)

## Format (Assembler)

Operation	Operanden
PRNTDOC	MF=C/D/E/L/M  ,PREFIX= <u>S</u> / <name 1..1>  ,MACID= <u>CPA</u> / <name 1..3>  ,PARAM=<name 1..27>  ,VARIANT= <u>001</u> / <c-string 3..3>  ,CALLER= <u>*USER</u> / *SYSTEM  ,RSOPAR= <u>*NONE</u> / <var: pointer> / (<reg: pointer>)  ,SPSPAR= <u>*NONE</u> / <var: pointer> / (<reg: pointer>)  ,DPPAR= <u>*NONE</u> / <var: pointer> / (<reg: pointer>)  ,DSEMPAR= <u>*NONE</u> / <var: pointer> / (<reg: pointer>)  ,DTYPE= <u>*FILE</u> / *LIBRARY_ELEMENT / *POSIX / *EAM / *SYSTEM_FILE / <var: enum-of dtype_set>  ,DNAME=( <i>pointer, length</i> ) <i>pointer</i> : <u>*NONE</u> / <var: pointer> / (<reg: pointer>) <i>length</i> : <u>*STD</u> / <integer 1..1024> / <var: int: 2> / (<reg: int:2>)  ,EAMN=array(16): <u>*STD</u> / <integer 0..65535> / <var: int: 4> / (<reg: int:4>)  ,LSTNN=array(16): <u>*DUMMY</u> / <integer 0..101> / *SYSLST / *SYSOUT / <var: int: 2> / (<reg: int:2>)

(Teil 1 von 7)

Operation	Operanden
PRNTDOC	<p><b>,LIBELEM</b>=(<i>element, version, type</i>)  <i>element</i>: <u>*DUMMY</u> / &lt;var: char: 80&gt; / (&lt;reg: char:80&gt;) /  &lt;c-string: filename 1..80 with-wild&gt;  <i>version</i>: <u>*HIGHEST</u> / &lt;var: char 1:40&gt; / (&lt;reg: char:40&gt;) /  &lt;c-string: filename 1..40 with-wild&gt;  <i>type</i>: &lt;var: char: 12&gt; / (&lt;reg: char:12&gt;) /  &lt;c-string: alphanum-name 1..12 with-wild&gt;</p> <p><b>,SECTREC</b>=(<i>ident, length, position</i>)  <i>ident</i>: <u>*NONE</u> / &lt;c-string 1..60&gt; / &lt;var: char: 60&gt; / (&lt;reg: char:60&gt;)  <i>length</i>: <u>0</u> / &lt;integer 1..60&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  <i>position</i>: <u>*STD</u> / &lt;integer 1..2047&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)</p> <p><b>,FIRSREC</b>=(<i>begin, ident, length, position, occur</i>)  <i>begin</i>: <u>*BEGIN_OF_FILE</u> / <u>*BY_STRING_ID</u> /  &lt;integer 1..2147483647&gt; / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;)  <i>ident</i>: <u>*DUMMY</u> / &lt;c-string 1..60&gt; &lt;var: char: 60&gt; / (&lt;reg: char:60&gt;)  <i>length</i>: <u>0</u> / &lt;integer 1..60&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  <i>position</i>: <u>*STD</u> / &lt;integer 1..2047&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)  <i>occur</i>: <u>1</u> / &lt;integer 1..32767&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)</p> <p><b>,LASTREC</b>=(<i>begin, ident, length, position, occur</i>)  <i>begin</i>: <u>*END_OF_FILE</u> / <u>*BY_STRING_ID</u> /  &lt;integer 1..2147483647&gt; / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;)  <i>ident</i>: <u>*DUMMY</u> / &lt;c-string 1..60&gt; / &lt;var: char: 60&gt; /  (&lt;reg: char:60&gt;)  <i>length</i>: <u>0</u> / &lt;integer 1..60&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  <i>position</i>: <u>*STD</u> / &lt;integer 1..2047&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)  <i>occur</i>: <u>1</u> / &lt;integer 1..32767&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)</p> <p><b>,RECPART</b>=(<i>first, last</i>)  <i>first</i>: <u>1</u> / &lt;integer 1..32767&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)  <i>last</i>: <u>*STD</u> / &lt;integer 1..32767&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)</p>

(Teil 2 von 7)

Operation	Operanden
PRNTDOC	<pre> ,LINESP=(spacing, position)   spacing: *STD / *SPACE_1 / *SPACE_2 / *SPACE_3 /            *BY_ASA_CONTROL / *BY_EBCDIC_CONTROL /            *BY_IBM_CONTROL / *NO / &lt;var: enum-of space_set:1&gt;   position: *STD / &lt;integer 1..2040&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)  ,DOCFORM=*TEXT / *PAGE_FORMAT / *SPECIAL_FORMAT /          &lt;var: enum-of docform_set:1&gt;  ,CONTMOD=(mode, pcc, type)   mode: *STD / *PHYSICAL / *LINE_MODE /         *LOGICAL / *APA / *PAGE_MODE /         &lt;var: enum-of control_mode_set:1&gt;   pcc: *DUMMY / *YES / *NO / &lt;var: enum-of pcc_set:1&gt;   type: *DUMMY / *COMPATIBLE / *HP /         &lt;var: enum-of control_type_set:1&gt;  ,ADDCOP=<u>Q</u> / &lt;integer 0..255&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  ,FAMILY=*STD / *YES / *NO / &lt;var: enum-of family_set:1&gt;  ,CHECKP=*ON_PAGES / *ON_SECTION_RECORDS /         &lt;var: enum-of checkp_set:1&gt;  ,STPROC=*IMMEDIATE / *NO / *AT_FILE_CLOSING /         &lt;var: enum-of stdproc_set:1&gt;  ,MONJV=*NONE / *STD / &lt;var: char: 54&gt; / (&lt;reg: char:54&gt;) /         &lt;c-string: filename 1..54 with-wild&gt;  ,JVPASSW=*NONE / &lt;var: char: 4&gt; / (&lt;reg: char:4&gt;) /         &lt;c-string: c-string with-low 1..4&gt;  ,PRJNAME=*JOB_NAME / &lt;var: char: 8&gt; / (&lt;reg: char:8&gt;) /         &lt;c-string: c-string with-low 1..8&gt; </pre>

(Teil 3 von 7)

Operation	Operanden
PRNTDOC	<pre> ,PRJPRIO=<u>*JOB_PRIORITY</u> / &lt;integer 30..255&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  ,PROCADM=(<i>user, account, passwd</i>)   <i>user</i>: <u>*SAME</u> / &lt;var: char: 8&gt; / (&lt;reg: char:8&gt;) /     &lt;c-string: alphanum-name 1..8, c-string 1..8&gt;   <i>account</i>: <u>*NONE</u> / &lt;var: char: 8&gt; / (&lt;reg: char:8&gt;) /     &lt;c-string: alphanum-name 1..8, c-string 1..8&gt;   <i>passwd</i>: <u>*NONE</u> / &lt;var: char: 8&gt; / (&lt;reg: char:8&gt;) /     &lt;c-string: c-string 1..8&gt;  ,TRUNC=<u>*STD</u> / *DELETE_FILE / *KEEP_FILE /   &lt;var: enum-of truncation_set:1&gt;  ,LOCKF=<u>*STD</u> / *YES / *NO / &lt;var: enum-of lockf_set:1&gt;  ,DELF=<u>*NO</u> / *YES / *DESTROY / &lt;var: enum-of delf_set:1&gt;  ,OUTPART=(<i>begin, end, dimension</i>)   <i>begin</i>: <u>*BEGIN_OF_FILE</u> / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;) /     &lt;integer -2147483647..2147483647&gt; /   <i>end</i>: <u>*END_OF_FILE</u> / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;) /     &lt;integer 1..2147483647&gt; /   <i>dimension</i>: <u>*PAGES</u> / *LINES /     &lt;var: enum-of layout_part_dim_set:1&gt;  ,PAGECOP=<u>*STD</u> / &lt;integer 0..255&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)  ,LEFTMAR=<u>*STD</u> / &lt;integer 0..31&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  ,LINEPP=<u>*STD</u> / &lt;integer 1..32767&gt; / &lt;var: int: 2&gt; / (&lt;reg: int:2&gt;)  ,HEADLIN=<u>*NO</u> / *STD / *DATE / *FRECORD /   *PAGE / *DATE_FRECORD / *DATE_PAGE /   *FRECORD_PAGE / *DATE_FRECORD_PAGE /   &lt;var: enum-of headline_set:1&gt; </pre>

(Teil 4 von 7)

Operation	Operanden
PRNTDOC	<pre> ,OUTFORM=<u>CHARACTER</u> / *HEXADECIMAL /     &lt;var: enum-of outform_set:1&gt;  ,TWOSIDE=<u>STD</u> / *NO / *YES / *TUMBLE /     &lt;var: enum-of twoside_set:1&gt;  ,ROT=<u>NO</u> / *ROT_0 / *ROT_90 / *ROT_180 / *ROT_270 /     *ROT_0_180 / *ROT_90_270 / *ROT_180_0 /     *ROT_270_90 / *BY_CONTROL_CODES /     &lt;var: enum-of rotation_set:1&gt;  ,INTRAY=<u>STD</u> / &lt;integer 1..99&gt; / *IGNORE / &lt;var: int: 1&gt; /     *BY-FORMAT  ,INTRAYF=&lt;var: enum-of _inray_format_set:1&gt; /*A3 / *A4 / *A5 /     *B4 / *B5 / *FOLIO / *INVOICE / *EXEC /     *LEGAL / *LETTER / *DOUBLE-LETTER / *MONARCH /     *COMMERCIAL-10 / *DL / *C5 / *MANUAL / *A3_UNCUT /     *A4_UNCUT / LEDGER  ,OUTTRAY=<u>STD</u> / &lt;integer 1..99&gt; / &lt;var: int: 1&gt; / *IGNORE /     *SORTER  ,COVPAGE=(<i>text, header-exit, trailer-exit</i>)     <i>text:</i> <u>NONE</u> / &lt;var: char: 32&gt; / (&lt;reg: char:32&gt;) /     &lt;c-string: c-string 1..32&gt;     <i>header-exit:</i> <u>NO</u> / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;) /     &lt;integer 0..2147483639&gt;     <i>trailer-exit:</i> <u>NO</u> / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;) /     &lt;integer 0..2147483639&gt; /  ,FORM=<u>STD</u> / &lt;var: char: 6&gt; / (&lt;reg: char:6&gt;) /     &lt;c-string: alphanum-name 1..6&gt;  ,LOOP=<u>STD</u> / &lt;var: char: 3&gt; / (&lt;reg: char:3&gt;) /     &lt;c-string: alphanum-name 1..3&gt; </pre>

(Teil 5 von 7)



Operation	Operanden
PRNTDOC	<pre> ,ROTLOOP=<u>*STD</u> / &lt;var: char: 3&gt; / (&lt;reg: char:3&gt;) /       &lt;c-string: alphanum-name 1..3&gt;  ,CHARSET=array(16): <u>*STD</u> / &lt;var: char: 3&gt; / (&lt;reg: char:3&gt;)/       &lt;c-string: alphanum-name 1..3&gt;  ,POOLNAM=<u>*NONE</u> / &lt;var: char: 4&gt; / (&lt;reg: char:4&gt;) /       &lt;c-string: name 1..4&gt;  ,POOLIND=<u>0</u> / &lt;integer 0..64&gt; / &lt;var: int: 1&gt; / (&lt;reg: int:1&gt;)  ,EFO=<u>*NONE</u> / &lt;var: char: 2&gt; / (&lt;reg: char:2&gt;) /       &lt;c-string: c-string 2..2&gt;  ,OVERLAY=(<i>face, reverse</i>)   <i>face</i>: <u>*STD</u> / <u>*NONE</u> / &lt;integer 1..127&gt; / &lt;var: int: 1&gt; /         (&lt;reg: int:1&gt;)   <i>reverse</i>: <u>*STD</u> / <u>*NONE</u> / &lt;integer 1..127&gt; / &lt;var: int: 1&gt; /           (&lt;reg: int:1&gt;)  ,FOB=<u>*NONE</u> / &lt;var: char: 4&gt; / (&lt;reg: char:4&gt;) /       &lt;c-string: alphanum-name 1..4, c-string 1..4&gt;  ,PAGEPCL=<u>*STD</u> / &lt;integer 1..50000&gt; / &lt;var: int: 4&gt; / (&lt;reg: int:4&gt;)  ,USERRES=<u>*DUMMY</u> / &lt;var: char: 44&gt; / (&lt;reg: char:44&gt;)       &lt;c-string: filename 1..44 without-userid, c-string 1..44&gt;  ,TRANTAB=(<i>name, file</i>)   <i>name</i>: <u>*NONE</u> / &lt;var: char: 8&gt; / (&lt;reg: char:8&gt;) /         &lt;c-string: alphanum-name 1..8, c-string 1..8&gt;   <i>file</i>: <u>*SYSTEM</u> / <u>*STD</u> / &lt;var: char: 44&gt; / (&lt;reg: char:44&gt;) /         &lt;c-string: filename 1..44 without-userid, c-string 1..44&gt;  ,PRNAME = <u>*STD</u> / &lt;c-string: c-string 1..8&gt; / &lt;var: char 1..8&gt; / *IPP </pre>

(Teil 6 von 7)

Operation	Operanden
PRNTDOC	<pre> ,PRSTYPE=<u>ANY</u> / *LP65_PRINTER / *HP_PRINTER / *APA_PRINTER / &lt;var: enum-of prtype_set:1&gt;  ,SRTMODE=<u>NO</u> / &lt;var: enum-of_srtmode_set:1&gt; / *GROUP / *COLLATE / *STACKER / *AUTOMATIC  ,VIRTUAL=<u>STD</u> / &lt;var: enum-of_virtual_set:1&gt; / *ALLOWED / *NOT-ALLOWED / *MUST  ,PROGRAM=(<i>name, string</i>)   <i>name</i>: <u>ANY</u> / &lt;var: char: 8&gt; /     &lt;c-string: alphanum-name 1..8, c-string 1..8&gt;   <i>string</i>: <u>NONE</u> / &lt;var: char: 32&gt; /     &lt;c-string: alphanum-name 1..32, c-string 1..32&gt;  ,PRJCLAS=<u>JOB-CLASS</u> / &lt;var: int: 1&gt; / &lt;integer 1..255&gt;  ,FRMNAME=<u>STD</u> / &lt;c-string: c-string 1..63&gt; / &lt;var: char: 63&gt; /   (&lt;reg: char:63&gt;)  ,TOPOFF=<u>IGNORE</u> / &lt;integer -255..255&gt;  ,LEFTOFF=<u>IGNORE</u> / &lt;integer -255..255&gt;  ,SCHEDTIME = (<i>schdate, schtime</i>)   <i>schdate</i>: <u>TODAY</u> / &lt;var: char 1..10&gt; / &lt;c-string 10..10&gt;   <i>schtime</i>: <u>NOW</u> / &lt;var: char 1..5&gt; / &lt;c-string 5..5&gt;  ,NOTIFPAR = &lt;var: pointer&gt; / <u>NONE</u> </pre>

(Teil 7 von 7)

## Operandenbeschreibung

### **ADDCOP=0 / <integer 0..255> / <var: int: 1> / (<reg: int:1>)**

Legt fest, wie oft die Datei zusätzlich auszudrucken ist. Jeder zusätzliche Ausdruck erhält ein eigenes Deckblatt. Als Wert dieses Operanden kann eine Ganzzahl von 0 bis 255 oder der Name eines Feldes, das mit FL definiert ist (Länge 1 Byte) bzw. ein Register, in dem der Wert steht, angegeben werden.

Voreinstellung: 0; kein zusätzlicher Ausdruck.

### **CALLER=**

Aufrufer des Makros.

### **CALLER=\*USER**

Aufrufer ist Anwender (TU).

### **CALLER=\*SYSTEM**

Aufrufer ist System (TPR).

### **CHARSET=array(16): \*STD / <var: char: 3> / (<reg: char:3>) / <c-string: alphanum-name 1..3>**

Namen der Fonts, die für die Ausgabe zu verwenden sind.

In der Liste können für lokalen SPOOL maximal 4, für RSO maximal 16 Fonts angegeben werden. Die Fonts müssen für lokalen SPOOL in der Ressourcen-Bibliothek \$SYSSPOOL.PRFILE oder in einer Anwender-PRFILE (die mit dem Operanden USERRES anzugeben ist) enthalten sein.

Werden mehr als 4 Fonts benötigt, dann muss ein Font-Pool (Operand POOLNAM) angegeben werden.

Die Verwendung mehrerer Fonts in einem SPOOL-OUT-Auftrag setzt die Angabe CONTMOD=\*PAGE\_MODE voraus. Bei DOCFORM=\*TEXT wird nur der erste angegebene Font zum Drucken der (gesamten) Datei verwendet. Der Name des ersten Fonts und die Anzahl der angegebenen Fonts wird in der Ausgabe für das Kommando SHOW-PRINT-JOB-STATUS JOB-IDENTIFICATION=TSN(TSN=...) angezeigt.

Nur FUJITSU-SIEMENS-Standardfonts dürfen einen Namen bestehend aus drei numerischen Zeichen tragen.

Ist der Operand HEADLIN angegeben, so wird für die Überschriftzeile der erste angegebene Font verwendet.

Die Voreinstellung für CHARSET für das verwendete Formular kann mit SHOW-SPOOL-FORMS abgefragt werden; die Information ist dem Ausgabefeld C-S zu entnehmen.

Bei HP-Druckern ist zu beachten:

Die Systembetreuung kann einstellen, ob das Deckblatt mit dem Standardfont oder mit dem bei CHARSET angegebenen Font gedruckt wird. Diese Voreinstellung kann mit dem Kommando bzw. der SPSEVE-Anweisung SHOW-SPOOL-PARAMETERS abgefragt werden; die Information ist dem Ausgabefeld HEADER-PAGE: CHARACTER-SET=... zu entnehmen.

**CHARSET=\*STD**

Aus dem gewünschten Formular wird der Standardfont für diesen Druckertyp gewählt. Er kann mit SHOW-SPOOL-FORMS abgefragt werden.

**CHARSET=<c-string: alphanum-name 1..3>**

Namen der Fonts, mit denen der SPOOL-OUT-Auftrag abgearbeitet werden soll. Die Zeichenfolge ist in Hochkommata einzuschließen.

Bei DOCFORM=\*TEXT (Voreinstellung) wird nur der erste angegebene Font zum Drucken verwendet.

Für Laserdrucker im lokalen SPOOL-Betrieb können maximal 4 Fonts angegeben werden. Die Druckersteuerzeichen zur Fontumschaltung werden nur ausgewertet, wenn zugleich CONTMODE=\*PAGE\_MODE angegeben wird.

Für RSO-Drucker können maximal 16 Fonts angegeben werden. Fontkennzeichen (CSI) im Text werden nur ausgewertet, wenn zugleich CONTMOD=\*LOGICAL angegeben wird. Der Makro wird abgewiesen, wenn in Verbindung mit DOCFORM=\*TEXT eine Liste mit Fonts angegeben wird.

**CHARSET=<var: char: 3> / (<reg: char:3>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 3 Byte), die als Name des Fonts interpretiert wird, hinterlegt.

**CHECKP=**

Legt fest, ob die Checkpoint-Verarbeitung durch den Controller auf der Basis von Seiten oder von Abschnitten (SECTIONS) durchgeführt werden soll.

**CHECKP=\*ON PAGES**

Standard-Wiederaufsetzverfahren.

Die Verarbeitung wird - bei Wiederanlauf eines unterbrochenen Jobs - eine gegebene Anzahl von Seiten zurückversetzt wieder aufgenommen.

**CHECKP=\*ON\_SECTION\_RECORDS**

Der Operandenwert kann für alle Druckertypen angegeben werden, bietet aber vor allem bei Ausgabe auf Drucker der Typen HP90 im TWO-UP-Verfahren und LP65 Vorteile:

Eine physische Seite kann bei diesem Druckertyp mehrere logische Seiten umfassen, ohne dass SPOOL dies feststellen kann (Information steht in der PCL-Datei), d.h. das an logischen Seiten orientierte Standard-Wiederaufsetzverfahren ist bei diesem Druckertyp sehr fehleranfällig.

SECTION-Sätze dienen hier als Markierungen zum Wiederaufsetzen. Sie teilen mit Hilfe von SECTION-Sätzen Ihre Datei in verschiedene Abschnitte ein. Diese SECTION-Sätze müssen die zu einer korrekten Verarbeitung der Daten notwendigen Druckerbefehle enthalten. Sind in einer physischen Seite mehrere logische Seiten enthalten, muss im SECTION-Satz zudem der Beginn einer physischen Seite deutlich gekennzeichnet sein.

Nach Auftreten eines Fehlers wird bei HOLD-PRINT-JOB und RESUME-PRINT-JOB die Verarbeitung eine gegebene Anzahl von Abschnitten (SECTIONS) weiter vorne in der Datei wieder aufgenommen; d.h. die Angabe bei RESTART-POSITION=\*PAGE(...) bzw.

\*BACK(...) bezeichnet hier nicht eine bestimmte Anzahl von Seiten, sondern von Abschnitten (SECTIONS). Auch die in Fehlermeldungen ausgegebenen Zahlen sind keine Seitenangaben, sondern Abschnittsangaben.

Um zu einem korrekten Ergebnis zu kommen, muss ein Abschnitt mindestens mit einer physischen Seite korrespondieren, im Idealfall genau einer Seite. Wird zusammen mit CHECKP=\*ON\_SECTION\_RECORDS der Operandenwert CONTMOD= \*PAGE\_MODE angegeben, haben Sie für Ausgabe auf Laserdrucker sicherzustellen, dass die SECTION-Sätze unmittelbar vor Datensätzen stehen, die die Steuerzeichenleiste am Anfang der Druckseite enthalten. Sobald neu positioniert wird (PRNTDOC oder Wiederanlauf eines unterbrochenen Jobs), muss der erste Datensatz, den SPOOL druckt, immer die Steuerzeichenleiste enthalten.

### CHECKP=<var: enum-of check\_set:1>

Die Vereinbarung bzgl. des Wiederanlaufs wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*ON_PAGES
1	*ON_SECTION_RECORDS
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

### CONTMOD=(mode, pcc, type)

Vereinbart, wie die Steuerzeichen interpretiert werden sollen (*mode*), ob die Steuerzeichenleiste am Anfang jeder Seite vorhanden sein muss (*pcc*) und ob die Steuerzeichen zur Verarbeitung auf Druckern des Typs HP bzw. HP90 geeignet (*type*) sind.

*mode*: \*STD / \*PHYSICAL / \*LINE\_MODE / \*LOGICAL / \*APA / \*PAGE-MODE /

<var: enum-of control\_mode\_set:1>

Vereinbart, wie die Steuerzeichen interpretiert werden sollen.

*mode*: \*PHYSICAL

Nur für RSO.

Es wird automatisch LINESP=\*NO gesetzt, d.h. Sie müssen Seiten- und Zeilenvorschub mit LINE-MODE-Steuerzeichen (d.h. VTSU-Codes, Druckersteuerzeichen, RENO-Kommandos) in der Datei selbst realisieren.

Die Angabe LINESP=\*BY\_EBCDIC\_CONTROL bei Datensätzen vom Typ D-2 bewirkt, dass das Vorschubsteuerzeichen im ersten Byte als Zeilen- oder Seitenvorschubsteuerzeichen ausgewertet wird. Es werden auch nichtabdruckbare Zeichen zum Drucker übertragen (im Unterschied zu CONTMOD=\*LINE\_MODE). Sie sind selbst verantwortlich für die Richtigkeit der Steuerzeichen in der Datei (einschließlich der Vorschubsteuerung).

**mode: \*LINE\_MODE**

Nur für RSO.

Es werden Datensätze vom Typ C (siehe Anhang) ausgedruckt. Die Datensätze können die Daten gemischt mit LINE-MODE-Steuerzeichen enthalten. LINE-MODE-Steuerzeichen sind (in beliebiger Kombination)

- Druckersteuerzeichen (d.h. physikalische Steuerzeichen beginnend mit X'27' oder X'3C')
- RENO-Kommandos
- VTSU-Codes.

Die Kontrolle über Satz- und Dateiaufbau (einschließlich Seiten- und Zeilenvorschub über LINE-MODE-Steuerzeichen) liegt allein beim Anwender. Er muss beim Seitendrucker 9025/9026-RENO auch für die richtige Anfangspositionierung auf dem Papier sorgen.

Ein Loop wird nicht ausgewertet. Nichtabdruckbare Zeichen, d.h. Zeichen mit einem sedezimalen Wert < X'40' werden als Leerzeichen ausgegeben.

Der SPOOLOUT-Auftrag wird mit dem Standardfont des verwendeten Formulars ausgeführt, bis Sie den Font innerhalb der Datei mittels LINE-MODE-Steuerzeichen wechseln.

**mode: \*LOGICAL**

Nur für RSO-Drucker.

Gibt an, dass Datensätze vom Typ B-1 oder B-2 auszudrucken sind, d.h. Datensätze, die außer einem Vorschubsteuerzeichen im ersten Byte auch Daten gemischt mit Fontkennzeichen, Druckersteuerzeichen, RENO-Kommandos und VTSU-Codes enthalten können (siehe Handbuch „[RSO \(BS2000/OSD\)](#)“, Abschnitt Datensatztyp Typ B-1/B-2). Mit Ausnahme der VTSU-Codes VPA, NP, VT, NL und CR, die als Leerzeichen ausgegeben werden, werden die oben genannten Steuerzeichen ausgewertet.

Ein Fontkennzeichen, ein VTSU-Code oder ein RENO-Kommando bleibt gültig, bis ein neues Steuerzeichen angegeben wird.

Bei fehlenden Fontkennzeichen wird mit dem Standardfont des Formulars gedruckt. Da der Seitenvorschub bei CONTMOD=\*LOGICAL über einen Loop oder konstanten Zeilenvorschub realisiert wird, sollten auch die RENO-Kommandos \LF, \FF und \CR nicht in der Datei vorkommen. Auch das Setzen der Formularhöhe ist nicht erlaubt.

**mode: \*APA**

Gibt an, dass die auszudruckende Datei APA-Druckersteuerzeichen enthält, die ausgewertet werden sollen.

**mode: \*PAGE\_MODE**

Die Steuerzeichen sollen als spezifische Steuerzeichen für Seitendrucker interpretiert werden.

**mode: <var: enum-of control\_mode\_set:1>**

Die Vereinbarung, wie die Steuerzeichen interpretiert werden sollen, wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*STD
2	*PHYSICAL
4	*PAGE_MODE
8	*LOGICAL
16	*LINE_MODE
64	*APA
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**pcc: \*DUMMY / \*YES / \*NO / <var: enum-of pcc\_set:1>**

Legt fest, ob die Steuerzeichenleiste am Anfang jeder Seite (d.h. im Loop immer nach Sprung auf den Vertikaltabulator "Kanal 1" ) vorhanden sein muss. Der Operand wird nur bei Ausgabe auf HP und HP90-Druckern ausgewertet (Einstellung CONT-MODE=\*PAGE\_MODE).

**pcc: \*DUMMY**

Standardwert: Der Operand soll nicht ausgewertet werden.

**pcc: \*YES**

Die Steuerzeichenleiste muss vorhanden sein.

**pcc: \*NO**

Keine Steuerzeichenleiste am Seitenanfang. Dies bewirkt jedoch, dass bei Ausgabe auf HP-Drucker folgende Funktionen nicht gesteuert werden können:

- Film-Dia auf einzelnen Seiten der Datei; im Makro PRNTDOC angegeben, wird ein Film-Dia auf jeder Seite der Druckdatei des SPOOL-OUT-Auftrags verwendet.
- Seitenkopien für einzelne Seiten der Datei; alle Seiten der Druckdatei werden mit so vielen Kopien ausgegeben, wie im Makro PRNTDOC angegeben wurde.
- Spaltenweises Einrücken auf einzelnen Seiten; der im Makro PRNTDOC angegebene Wert gilt für alle Seiten der Druckdatei.
- FOB-Datendia auf einzelnen Seiten der Datei; im Makro PRNTDOC angegeben, wird ein FOB-Datendia auf jeder Seite der Druckdatei des SPOOL-OUT-Auftrags verwendet.
- Die Angabe einer Kopienreferenznummer ist nicht möglich.
- Steuerung der Seitendrehung für einzelne Seiten der Druckdatei; alle Seiten werden entweder im Hochformat oder im Querformat ausgegeben, so wie dies im Makro PRNTDOC angegeben ist.

***pcc: <var: enum-of pcc\_set:1>***

Die Vereinbarung bzgl. der Steuerzeichenleiste wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*YES
1	*NO
2	*DUMMY
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

***type: \*DUMMY / \*COMPATIBLE / \*HP / <var: enum-of control\_type\_set:1>***

Legt fest, ob die Steuerzeichen zur Verarbeitung auf Druckern des Typs HP bzw. HP90 geeignet sind oder in diese Form konvertiert werden müssen.

***type: \*DUMMY***

Der Operand soll nicht ausgewertet werden.

***type: \*COMPATIBLE***

In der Datei sind keine für Drucker des Typs HP bzw. HP90 spezifischen Steuerzeichen hinterlegt; die Steuerzeichen müssen von SPOOL in diese Form konvertiert werden.

***type: \*HP***

In der Datei sind spezifische Steuerzeichen für einen Drucker des Typs HP bzw. HP90 hinterlegt, die auch nur von Druckern dieses Typs verarbeitet werden können.

***type: <var: enum-of control\_type\_set:1>***

Die Vereinbarung bzgl. der Konvertierung der Steuerzeichen wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*DUMMY
1	*COMPATIBLE
2	*HP
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	



**COVPAGE=(text, header-exit, trailer-exit)**

Angaben für Exit-Routinen, die das Drucken von Deck- und Schlussblättern betreffen.

**text: \*NONE / <c-string 1..32> / <var: char: 32> / (<reg: char:32>)**

Legt fest, ob eine Zeichenfolge auf Deck- und Schlussblättern erscheinen soll.

**text: \*NONE**

Es soll keine Zeichenfolge auf Deck- und Schlussblättern erscheinen.

**text: <c-string 1..32>**

Die angegebene Information (max. 32 Zeichen) wird zur Verarbeitung von Exits im SCB gespeichert. Die ersten 8 Zeichen werden auf das Deckblatt als Großdruckzeile unter die Mailing Box gedruckt.

Nur Buchstaben, Ziffern und einige Sonderzeichen werden - als Großdruck - dargestellt, alle anderen Zeichen werden automatisch durch das abdruckbare Zeichen '?' für das Deckblatt ersetzt.

*Aufbau des Deckblattes*

1. Benutzerkennung in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
2. Abrechnungsnummer in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
3. Auftragsname in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
4. Mailing Box (Adress- und Identifikationsfeld; 12 Zeilen + 2 Leerzeilen)
5. 'text' in "Riesen-Buchstaben" (10 Zeilen + 2 Leerzeilen)

*Prioritätsreihenfolge*

1. Mailing Box (Adress- und Identifikationsfeld; 12 Zeilen + 2 Leerzeilen)
2. 'text' in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
3. Auftragsname in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
4. Benutzerkennung in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
5. Abrechnungsnummer in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)

**text: <var: char: 32> / (<reg: char:32>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 32 Byte), die als zu speichernde Information interpretiert wird, hinterlegt.

**header-exit: \*NO / <integer 0..2147483647> / <var: int: 4> / (<reg: int:4>)**

Nummer der gewünschten Deckblätter.

Die genaue Bedeutung des Operanden hängt jeweils von den Rechenzentrumspezifischen Vereinbarungen ab.

**header-exit: <var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Nummer der Deckblätter interpretiert wird, hinterlegt.

**trailer-exit: \*NO / <integer 0..2147483647> / <var: int: 4> / (<reg: int:4>)**

Nummer der gewünschten Schlussblätter.

Die genaue Bedeutung des Operanden hängt jeweils von den Rechenzentrumspezifischen Vereinbarungen ab.

**trailer-exit: <var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Nummer der Schlussblätter interpretiert wird, hinterlegt.

### **DELFL=**

Legt fest, ob die Datei nach Beendigung der Ausgabe gelöscht werden soll und wenn ja, ob zusätzlich ihr Katalogeintrag und die Daten mit X'00..0' überschrieben werden sollen.

Voreinstellung: Die Datei wird nach dem Drucken nicht gelöscht und nicht mit binären Nullen überschrieben. Sie müssen zum Schreibzugriff auf die Datei berechtigt sein.

Gehört die auszugebende Datei zu einer Dateigenerationsgruppe, wird der Operand DELFL ignoriert.

Sollen in einem SPOOLOUT-Auftrag mehrere Elemente einer PLAM-Bibliothek mit SECTREC=... ausgegeben werden, wird der Operand DELFL auf \*NO gesetzt (d.h. unterdrückt). Der Operand darf nicht zusammen mit \*SYSLST, \*SYSLSTnn oder \*SYSOUT angegeben werden.

### **DELFL=\*NO**

Die Datei soll nach dem Drucken nicht gelöscht werden (Ausnahme: EAM- und Systemdateien).

### **DELFL=\*YES**

Die Datei soll gelöscht werden, sobald die Ausgabe beendet ist.

### **DELFL=\*DESTROY**

Nicht für EAM- und katalogisierte Systemdateien.

Legt fest, dass nach dem Drucken der Datei sowohl ihr Katalogeintrag als auch die Daten selbst mit binär Null überschrieben werden sollen.

### **DELFL=<var: enum-of delf\_set:1>**

Die Vereinbarung bzgl. des Löschens der Dateien wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*YES
2	*DESTROY
3	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**DNAME=(pointer, länge)**

Vereinbart den Namen der auszudruckenden katalogisierten Datei und die Länge des Dateinamens.

**pointer: \*NONE / <var: pointer> / (<reg: pointer>)**

Vereinbart den Namen der auszudruckenden katalogisierten Datei.

**pointer: \*NONE**

Es soll keine katalogisierte Datei, sondern ein Bibliothekselement oder eine Systemdatei ausgedruckt werden.

**pointer: <var: pointer> / (<reg: pointer>)**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht der benötigte Wert selbst, sondern die Adresse eines Speicherplatzes, an dem der Wert hinterlegt ist (A(feld) oder Angabe eines Registers).

**länge: \*STD / <integer 1..1024> / <var: int: 2> / (<reg: int:2>)**

Vereinbart die Länge des Dateinamens.

**länge: \*STD**

Die Länge des Dateinamens beträgt 54 Zeichen.

**länge: <integer 1..1024>**

Angabe eines ganzzahligen Wertes für die Länge des Dateinamens.

**länge: <var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Länge des Dateinamens interpretiert wird, hinterlegt.

*Beispiele*

PRNTDOC DNAME=(A(VAR1),41)

Der Name der auszudruckenden Datei wird im Feld VAR1 hinterlegt. Von diesem Feld sind 41 Zeichen als Dateinamenslänge auszuwerten.

PRNTDOC DNAME=(A(VAR1))

Der Name der auszudruckenden Datei wird im Feld VAR1 hinterlegt. Von diesem Feld sind 54 Zeichen (Voreinstellung) als Dateinamenslänge auszuwerten.

PRNTDOC DNAME=(A(VAR1),VAR2)

Der Name der auszudruckenden Datei wird im Feld VAR1 hinterlegt. Die Länge des Dateinamens ist im Feld mit dem Namen VAR2 hinterlegt.

PRNTDOC DNAME=(\*NONE,\*STD)

Voreinstellung: Es soll keine katalogisierte Datei ausgedruckt werden, die Standard-Längenangabe wird ignoriert.

**DOCFORM=**

Legt fest, welcher Art die Dokumentinhalte sind, d.h. welches Format die auszudruckende Datei bezüglich der Auswertung von Vorschubsteuerzeichen, Druckersteuerzeichen, Fontkennzeichen, RENO-Kommandos und/oder VTSU-Codes aufweist.

**DOCFORM=\*TEXT**

Die Datei weist bis auf Vorschubsteuerzeichen keine druckerspezifischen Steuerzeichen auf. Im Operand LINESP kann die Position des Vorschubsteuerzeichens im Datensatz angegeben werden.

Die Daten werden ohne irgendeine Veränderung an den Drucker geschickt. Da nur die auszudruckenden Daten an den Drucker geschickt werden, können Datensätze „abgeschnitten“ werden.

Für HP- und HP90-Drucker gilt:

Bei Angabe des Operanden PRTYPE=\*ANY wird das Zeichen X'FF' durch das Zeichen X'1F' ersetzt, um kompatibel zu sein mit der PRM-Anweisung CONVERT-PRINT-RESOURCES.

Für LP65-Drucker gilt:

SPOOLOUT-Aufträge, bei denen DOCFORM=\*TEXT spezifiziert ist, können beliebige LP65-Steuerzeichen und Druckersteuerzeichen enthalten. Nur Sätze, die länger als 8192 Zeichen sind, werden abgeschnitten.

Für RSO gilt:

Datensätze der Typen A-1 oder A-2 sollen ausgedruckt werden (keine Steuerzeichen im Datenstrom), d.h. außer Vorschubsteuerzeichen in der ersten Spalte der Datensätze werden keine Steuerzeichen ausgewertet. Aus diesem Grunde ist auch kein Fontwechsel möglich, d.h. es wird mit dem Standardfont des Formulars oder, falls angegeben, mit dem ersten bei CHARSET angegebenen Font ausgedruckt. Nichtabdruckbare Zeichen, d.h. Zeichen mit einem sedezimalen Wert < X'40', werden als Leerzeichen ausgegeben.

Übersteigt die Satzlänge die max. Zeilenlänge, so wird der Datensatz beim Ausdruck abgeschnitten. Die max. Zeilenlänge ist abhängig vom Zeichenabstand; dieser ist durch den verwendeten Font (siehe Operand CHARSET) definiert.

**DOCFORM=\*PAGE\_FORMAT**

Die Datei weist spezifische Steuerzeichen für Laserdrucker auf. Mit dem Operanden CONTMOD kann festgelegt werden, wie die Steuerzeichen interpretiert werden sollen.

**DOCFORM=\*SPECIAL\_FORMAT**

Im Dokument wird eine druckerspezifische Sprache verwendet. In diesem Fall stellen die Subsysteme SPOOL und Distributed Print Services (Dprint) eine transparente Steuerung für das Dokument zur Verfügung, d.h. die Datei wird ohne Auswertung an einen RSO- oder Xprint-Drucker übertragen. Der Wert des Operanden FRMNAME legt fest, welches Format das Dokument genau enthält. Siehe Operand FRMNAME auf [Seite 67](#).

**DOCFORM=<var: enum-of docform\_set:1>**

Die Vereinbarung bzgl. der Art des Dokumentinhaltes wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*TEXT
1	*PAGE_FORMAT
2	*SPECIAL_FORMAT
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**DPPAR=**

Es kann die Adresse einer Parameterliste angegeben werden, in der Distributed Print Services-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**DPPAR=\*NONE**

Die Werte sollen nicht über eine Parameterliste versorgt werden.

**DPPAR=<var: pointer> / (<reg: pointer>)**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die Dprint-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die Dprint-Parameterliste hinterlegt ist (A(feld) oder Angabe eines Registers).

**DSEMPAR=**

Es kann die Adresse einer Parameterliste angegeben werden, in der DSEM-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**DSEMPAR=\*NONE**

Die Werte sollen nicht über eine Parameterliste versorgt werden.

**DSEMPAR=<var: pointer> / (<reg: pointer>)**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die DSEM-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die DSEM-Parameterliste hinterlegt ist (A(feld) oder Angabe eines Registers).

**DTYPE=**

Typ der auszugebenden Dateien.

**DTYPE=\*FILE**

Es soll eine katalogisierte Datei ausgedruckt werden.

**DTYPE=\*LIBRARY\_ELEMENT**

Es soll ein Element aus einer PLAM-Bibliothek ausgedruckt werden.

**DTYPE=\*POSIX**

Es soll eine Datei aus einem POSIX-Dateisystem ausgedruckt werden.

**DTYPE=\*EAM**

Es soll eine temporäre Bindemoduldatei ausgedruckt werden.

**DTYPE=\*SYSTEM\_FILE**

Es soll eine Systemdatei (SYSLST oder SYSOUT) ausgedruckt werden.

**DTYPE=<var: enum-of dtype\_set:1>**

Die Vereinbarung bzgl. des Typs der auszugebenden Dateien wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*FILE
2	*LIBRARY_ELEMENT
3	*POSIX
4	*EAM
5	*SYSTEM_FILE
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**EAMN=array(16): \*STD / <integer 0..65535> / <var: int: 4> / (<reg: int:4>)**

Vereinbart die Nummern der auszugebenden EAM-Bindemoduldateien. Insgesamt können 16 Nummern angegeben werden.

Eine ausführliche Beschreibung von EAM-Dateien ist zu finden im Handbuch „[Einführung in das DVS](#)“.

**EAMN=\*STD**

Es soll die temporäre Bindemoduldatei des laufenden Auftrags ausgegeben werden.

**EAMN=<integer 1..65535>**

Die Nummer der EAM-Datei, die auszugeben ist, wird direkt durch Angabe der Ganzzahl angegeben.

**EAMN=<var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Nummer der EAM-Datei interpretiert wird, hinterlegt.

**EFO=**

Legt fest, ob ein Filmdia - für HP90- und 3365 Drucker ein EFO-Datendia - zu verwenden ist, wenn ein SPOOLOUT-Auftrag verarbeitet werden soll.

**EFO=\*NONE**

Es wird kein Filmdia (HP90 und 3365: kein EFO-Datendia) zur Ausgabe verwendet.

**EFO=<c-string 2..2>**

Name des Filmdias (HP90 und 3365: EFO-Datendias), das zur Verarbeitung des SPOOLOUT-Auftrags zu verwenden ist (der Name muss mit der Systembetreuung vereinbart worden sein).

**EFO=<var: char: 2> / (<reg: char:2>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 2 Byte), die als Name des Filmdias interpretiert wird, hinterlegt.

**FAMILY=**

Legt fest, ob bei Angabe von mehreren Dateien oder Bibliothekselementen in einem Makro PRNTDOC eine gemeinsame TSN vergeben wird (um sicherzustellen, dass diese Dateien nacheinander auf denselben Drucker ausgegeben werden).

**FAMILY=\*STD**

Für SPOOLOUT-Aufträge auf lokale Drucker und RSO-Geräte soll die Voreinstellung aus der SPOOL-Parameterdatei gelten, die für lokale und RSO-Drucker getrennt (und somit auch unterschiedlich) in der SPERVE-Anweisung MODIFY-SPOOL-PARAMETERS festgelegt werden kann. Sie können den Wert mit dem Kommando bzw. der SPERVE-Anweisung SHOW-SPOOL-PARAMETERS abfragen (Feld FAMILY-PROCESS).

**FAMILY=\*YES**

Bei Angabe von mehreren Dateien oder Bibliothekselementen in einem SPOOLOUT-Auftrag soll eine gemeinsame TSN vergeben werden (FAMILY-PRINT). Die einzelnen Dateien bzw. Bibliothekselemente können nicht parallel verarbeitet werden.

**FAMILY=\*NO**

Mehrere gleichzeitig in einem PRNTDOC angegebene Dateien oder Bibliothekselemente sollen jeweils unter einer eigenen TSN verarbeitet werden. Dadurch ist auch eine parallele Verarbeitung möglich.

**FAMILY=<var: enum-of family\_set:1>**

Die Vereinbarung bzgl. der Vergabe einer gemeinsamen TSN wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*STD
2	*YES
3	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**FIRSREC=(*begin, ident, length, position, occur*)**

Vereinbart, ob von der Eingabedatei nur eine bestimmte Anzahl von Datensätzen verarbeitet werden soll. Der Anfangs- und Enddatensatz kann durch eine Satznummer, eine Abschnittsnummer oder durch eine beliebige Zeichenfolge bestimmt werden. Mit dem Operanden LASTREC können die Angaben für den Enddatensatz eingestellt werden.

Mit "begin" kann ein Anfangsdatensatz, ab dem die Datei verarbeitet werden soll, angegeben werden.

Mit "ident" kann als Anfangsdatensatz ein Datensatz gewählt werden, in dem die angegebene Zeichenfolge auftritt.

Mit "length" kann die Länge des Datensatzes angegeben werden.

Mit "position" wird die Position innerhalb des Datensatzes, ab der der angegebene String beginnt, angegeben.

Mit "occur" kann vereinbart werden, ab dem wievielten Auftreten des Datensatzes, der den String enthält, die Ausgabe beginnen soll.

*begin*: **\*BEGIN OF FILE / \*BY\_STRING\_ID / <integer 1..2147483647> / <var: int: 4> / (<reg: int:4>)**

Vereinbart den Anfangsdatensatz, ab dem die Datei verarbeitet werden soll. Es kann hier der erste Datensatz der Datei, die Nummer eines Datensatzes oder einer Abschnittsmarke sowie eine beliebige Zeichenfolge in einem Datensatz gewählt werden.

*begin*: **\*BEGIN OF FILE**

Die Ausgabe beginnt mit dem ersten Datensatz der Datei, auch wenn SECTION-Sätze angegeben sind.

*begin*: **\*BY\_STRING\_ID**

Als Anfangsdatensatz wird ein Datensatz gewählt, in dem eine nachfolgend anzugebende Zeichenfolge auftritt.

*begin*: **<integer 1..2147483647>**

Der Anfangsdatensatz wird als Ganzzahl von 1 bis 2147483647 angegeben.

*begin*: **<var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Nummer des Anfangsdatensatzes interpretiert wird, hinterlegt.

*ident*: **\*DUMMY / <c-string 1..60> / <var: char: 60> / (<reg: char:60>)**

Vereinbart, ob die Ausgabe mit einem Datensatz beginnen soll, in dem ein bestimmter String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

*ident*: **\*DUMMY**

Es wird kein String abdruckbarer oder hexadezimaler Zeichen definiert.



***ident: <c-string 1..60>***

Die Ausgabe soll mit dem Datensatz beginnen, in dem der angegebene String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

***ident: <var: char: 60> / (<reg: char:60>)***

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 60 Byte), die als String interpretiert wird, hinterlegt.

***length: 0 / <integer 1..60> / <var: int: 1> / (<reg: int:1>)***

Gibt die Länge des Datensatzes an.

***length: 0***

Voreinstellung: Der Datensatz hat die Länge 0 Byte.

***length: <integer 1..60>***

Die Länge des Datensatzes wird als Ganzzahl von 1 bis 60 angegeben.

***length: <var: int: 1> / (<reg: int:1>)***

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Länge des Datensatzes interpretiert wird, hinterlegt.

***position: \*STD / <integer 1..2047> / <var: int: 2> / (<reg: int:2>)***

Legt fest, ab welcher Position (ab welchem Byte nach dem Satzlängelfeld) der angegebene String im SECTION-Satz beginnt.

***position: \*STD***

Der gesuchte String beginnt standardmäßig am Anfang des Datensatzes:

- bei einer SAM-Datei: mit dem ersten Byte nach dem Satzlängelfeld
- bei einer ISAM-Datei mit KEY-POS=5: mit dem ersten Byte nach dem Schlüssel
- bei einer ISAM-Datei mit KEY-POS > 5: mit dem ersten Byte nach dem Satzlängelfeld.

***position: <integer 1..2047>***

Angabe eines ganzzahligen Wertes für die Position des Strings im SECTION-Satz.

***position: <var: int: 2> / (<reg: int:2>)***

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Position des Datensatzes interpretiert wird, hinterlegt.

***occur: 1 / <integer 1..32767> / <var: int: 2> / (<reg: int:2>)***

Legt fest, ab dem wievielten Auftreten des Datensatzes, der den String enthält, die Ausgabe beginnen soll.

***occur: 1***

Die Ausgabe soll mit dem ersten Auftreten des Datensatzes beginnen.

**occur:** <integer 1..32767>

Ganzzahliger Wert für das Auftreten des Datensatzes.

**occur:** <var: int: 2> / (<reg: int:2>)

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Anzahl des Auftretens interpretiert wird, hinterlegt.

### **FOB=**

Legt fest, ob ein FOB-Datendia für die Verarbeitung des SPOOLOUT-Auftrags zu verwenden ist.

### **FOB=\*NONE**

Es wird kein FOB-Datendia zur Ausgabe verwendet.

**FOB='<alphanum-name 1..4>' / <c-string 1..4>**

Name des Dias, der zur Verarbeitung des SPOOLOUT-Auftrags verwendet werden soll. Bei Angabe eines Dias wird die Datei auf HP-PRINTER oder HP90-PRINTER ausgegeben. Die Verwendung eines Dias für den SPOOLOUT-Auftrag wird in der Ausgabe des Kommandos SHOW-PRINT-JOB-STATUS JOB-IDENTIFICATION=TSN(TSN=...) angezeigt.

**FOB=<var: char: 4> / (<reg: char:4>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 4 Byte), die als Name des Dias interpretiert wird, hinterlegt.

### **FORM=**

Bezeichnet das Papierformat, das für die Ausgabe verwendet werden soll (z.B. STD, STDSF1, STDWA4).

Für alle Druckertypen müssen Standardformate in der SPOOL-Parameterdatei definiert sein.

Mit SHOW-SPOOL-FORMS können Sie sich die Einträge auf SYSOUT ausgeben lassen. In der SPOOL-Parameterdatei ist auch festgelegt, ob Deck- und Schlussblätter gedruckt werden sollen.

### **FORM=\*STD**

Standardformular; Voreinstellung.

**FORM='<alphanum-name 1..6>'**

Name des Formulars, mit dem der SPOOLOUT-Auftrag verarbeitet werden soll. Implizit wird mit der Formularangabe ein Loop (oder eine Seiten- und Formatdefinition für APA-Drucker) benannt. Der zugeordnete Loop (bzw. die PAGEDEF und FORMDEF) muss in einer bestimmten Druckersteuerdatei enthalten sein. Die folgende Tabelle gibt Aufschluss darüber, welche Druckersteuerdatei für den jeweiligen Druckertyp diesen Loop (bzw. die PAGEDEF und FORMDEF) enthalten muss:

Ausgabegerät	Druckersteuerdatei mit dem Loop für das angegebene Formular
Drucker 3337, 3338, 3339, 3348, 3349, 3365, LP-EMULATED	\$\$SYSSPOOL.PRFILE
Drucker: 3351, 3353, 2090, 2140, 2240	\$\$SYSSPOOL.PRFILE oder die im Operanden USERRES angegebene Benutzer-PRFILE
APA-Drucker	\$\$SYSSPOOL.SYSPRT.SPS.021 oder die im Operanden USERRES angegebene Benutzer-SPSLIB

Der über den FORM-Operanden implizit benannte Loop wird ignoriert, wenn zugleich der Operand LOOP angegeben wird.

Ohne die Operanden FORM und LOOP wird mit dem für den jeweiligen Druckertyp eingetragenen Standardformular gedruckt.

Ein bei dem Operanden LOOP explizit angegebener Loop muss die gleiche Länge haben wie der dem verwendeten Formular zugeordnete Loop.

Bei APA-Druckern können keine Loops angegeben werden. Wenn Seiten- und Formatangaben im FORM-Operanden gemacht werden, werden diese zum Drucken von Header, Trailer und Meldungsseiten benutzt.

### **FORM=<var: char: 6> / (<reg: char:6>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 6 Byte), die als Name des Formulars interpretiert wird, hinterlegt.

### **FRMNAME=\*STD / <c-string: c-string 1..63> / <var: char: 63> / (<reg: char:63>)**

Angabe des Formats, das verarbeitet werden soll. Das Dokument wird im Transparent-Modus bearbeitet.

Name des Formats, in dem die auszugebenden Daten an den Drucker übergeben werden. Wenn kein Wert für FRMNAME angegeben wurde, wird automatisch ein Vorgabewert aus dem Wert des Operanden CONTMODE abgeleitet. Dabei ergibt sich für:

CONTMODE=\*PAGE-MODE(...)

FRMNAME=\*HP

CONTMODE=\*APA(...)

FRMNAME=\*SPDS

CONTMODE=\*LOGICAL / \*PHYSICAL / \*LINEMODE

FRMNAME=\*STD

Der Wert FRMNAME=\*STD bedeutet, dass der Inhalt der auszudruckenden Datei nicht relevant ist.

### **FRMNAME=\*STD**

Vorgabewert des Operanden.

**FRMNAME=<c-string: c-string 1..63>**

Der Name des Formats wird direkt angegeben.

**FRMNAME=<var: char: 63>**

Der Name des Formats wird indirekt über ein Feld angegeben.

**FRMNAME=(<reg: char:63>)**

Ein Register zeigt auf ein Feld, das den Namen des Formats enthält.

**HEADLIN=**

Legt fest, ob auf jede Seite (ausgenommen Deck- und Schlussblatt) eine Überschriftzeile zu drucken ist.

**HEADLIN=\*NO**

Eine Überschriftzeile wird nicht gedruckt.

**HEADLIN=\*STD**

Die Überschrift hat folgenden Aufbau:

Überschrift	DATE jjj-mm-tt	userid	datei	PAGE nnnn
Spalte	1	41(11,11)	60(21,67)	124 (77)

Die erste Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge < 132 und  $\geq 80$  Zeichen (mit abweichenden Werten). Die zweite Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge < 80 und  $\geq 70$  Zeichen (mit abweichenden Werten).

jj-mm-tt	Jahr-Monat-Tag
userid	Benutzerkennung
datei	Dateiname
nnnn	Seitennummer

Wenn nicht anders vereinbart, folgt der Überschriftzeile eine Leerzeile.

Die Überschrift verschiebt sich entsprechend der Angabe bei dem Operanden LEFTMAR, die Überschriftzeile wird jedoch ab Spalte 132 abgeschnitten.

DATE und PAGE sind in der Überschriftenzeile nur vorhanden, wenn gilt: Zeilengröße  $\geq 032$ .

**HEADLIN=\*DATE / \*FRECORD / \*PAGE / \*DATE\_FRECORD / \*DATE\_PAGE / \*FRECORD\_PAGE / \*DATE\_FRECORD\_PAGE**

Die Überschrift hat folgenden Aufbau:

HEADLIN=	*DATE	*FRECORD	*PAGE
Überschrift	DATE jjjj-mm-tt (jjjj-mm-tt)	erster Satz	PAGE nnnn (nnnn, nnnn)
Spalte	1	21	124 (77,67)

Bei \*DATE:

Die Angabe in Klammern gilt für eine Formulardefinition mit einer Zeilenlänge < 132 Zeichen und ≥ 70 (mit abweichenden Werten).

Bei \*PAGE

Die erste Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge < 132 und ≥ 80 Zeichen (mit abweichenden Werten). Die zweite Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge < 80 und ≥ 70 Zeichen (mit abweichenden Werten).

jjjj-mm-tt	Jahr-Monat-Tag
erster Satz	erster logischer Satz der Datei
nnnn	Seitennummer

Fehlt eine der Angaben \*DATE, \*FRECORD oder \*PAGE , wird der entsprechende Abschnitt mit Leerzeichen aufgefüllt.

Wenn nicht anders vereinbart, folgt der Überschriftzeile eine Leerzeile.

Die Überschrift verschiebt sich entsprechend der Angabe bei dem Operanden LEFTMAR, die Überschriftzeile wird jedoch ab Spalte 132 abgeschnitten.

\*DATE und \*PAGE sind in der Überschriftenzeile nur vorhanden, wenn gilt:  
Zeilengröße ≥ 032.

Bei Angabe von \*FRECORD wird der erste Satz nicht zum Datum zugehörig gesehen.  
Die Angaben können durch " \_ " (Unterstrich) verknüpft werden.

**HEADLIN=<var: enum-of headline\_set:1>**

Die Vereinbarung bzgl. der Überschriftzeile wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*NO
2	*STD
4	*FRECORD
8	*DATE
12	*DATE_FRECORD
16	*PAGE
20	*FRECORD_PAGE
24	*DATE_PAGE
28	*DATE_FRECORD_PAGE
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**INTRAY=**

Legt das Papiereingabefach für LP65-Drucker, PCL-Drucker, APA-Drucker und für die RSO-Drucker 2030-PCL, 4011, 4812, 4813, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-EPLQ, 9000-EPSQ, 9000-PCL, 9004, 9011, 9012, 9013, 9014, 9015, 9021, 9022, 9022-200, 9025, 9026-PCL, 9026-RENO, 9097 und DJET fest. Bei dem Druckertyp LP65 können die Fachnummern 1 bis 3, bei RSO-Druckern die Fachnummern 1 bis 99 angegeben werden.

**INTRAY=\*STD**

Im Fall, dass ein Druckauftrag an den lokalen SPOOL oder an einen speziellen Druckerpool adressiert wird, wird dieser Operand ignoriert. Andernfalls, wenn der Druckauftrag z. B. an einen bestimmten RSO-Drucker adressiert wird, wird Papier von dem Papiereingabefach verwendet, das im Geräteeintrag des LP65-Druckers in der SPOOL-Parameterdatei definiert ist. Die Auswahl des Papiereingabefaches ist gültig für den gesamten SPOOL-OUT-Vorgang, d.h. einschließlich der Verarbeitung von Deck- und Schlussblatt.

**INTRAY=\*IGNORE**

Nur für RSO.

Die Auswahl des Papiereingabefaches wird durch eine PCL-Datei oder über eine Eingabe an der Bedienungskonsole des Druckers gewählt. Die Prioritätsreihenfolge bei der Auswahl eines Papiereingabefaches ist wie folgt:

1. die im PRNTDOC angegebene Nummer des Papiereingabefaches
2. die eventuell erfolgte Auswahl über die PCL-Datei, die im Makro PRNTDOC angegeben wurde
3. die eventuell erfolgte Auswahl über die PCL-Datei, die in der Formulardefinition in der SPOOL-Parameterdatei festgelegt wurde.
4. die Standardwerte, die im Geräteeintrag definiert sind.

**INTRAY=<integer 1..99>**

Nummer des Einzugsfaches, aus dem das Papier sowohl für die Ausgabe der Datei selbst als auch zum Drucken des Deckblattes (HEADER-PAGE) und des Schlussblattes (TRAILER-PAGE) genommen wird.

Der angegebene Wert wird von RSO nicht überprüft. Mögliche Werte: 1...99.

Die folgende Tabelle zeigt den Druckertyp und die jeweils maximale Anzahl der Einzugsfächer (Spalte „max.“).

Drucker	max.	Drucker	max.	Drucker	max.	Drucker	max.
2030-PCL	2	8121	0	9002	0	9022	2
4011	2	9000	0	9003	0	9022-200	2
4812	2	9000-PCL	2	9004	3	9025	2
4813	2	9000-PRO	0	9011	2	9026-PCL	4
4818-PCL	2	9000-PS	0	9012	2	9026-RENO	4
4821-PCL	2	9000-EPFX	0	9013	3	9045-ANSI	0
4822-PCL	3	9000-EPLQ	2	9014	3	9046	0
4824-PCL	2	9000-EPSQ	2	9015	2	9645	0
4825-PCL	3	9001	0	9021	2	DJET	1
4830-PCL	3	9001-31	0				
4850-PCL	2	PCL-Drucker	3				

**INTRAY=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Nummer des Papiereingabefaches interpretiert wird, hinterlegt.

**INTRAY=\*BY-FORMAT**

Das Papiereingabefach wird über ein Schlüsselwort für das Format im Operanden INTRAYF festgelegt.

**INTRAYF=<var: enum-of\_intray\_format\_set: 1> / \*MANUAL / \*A3 / \*A4 / \*A5 / \*B4 / \*B5 / \*DOUBLE-LETTER / \*EXEC / \*FOLIO / \*INVOICE / \*LEGAL / \*LETTER / \*MONARCH / \*COMMERCIAL\_10 / \*DL / \*C5 / \*A3\_UNCUT / \*A4\_UNCUT / \*LEDGER**

Legt das Papiereingabefach über ein Schlüsselwort für das Format fest.

Um sicherzustellen, dass die im Operanden INTRAYF die angegebenen Werte für Papiergröße und Zeilenlänge die für das gewählte Papierformat zulässigen Maximalwerte nicht überschreiten, werden die Werte des INTRAYF-Operanden mit den zulässigen Maximalwerten für das Papierformat verglichen. Die folgende Tabelle enthält die für das jeweilige Papierformat zulässigen Maximalwerte.

Papierformat	max. PAGE-SIZE	max. LINE-SIZE
A3	165	116
A4	116	82
A5	82	58
B4	143	101
B5	101	71
FOLIO	129	85
INVOICE	85	55
EXEC	105	72
LEGAL	140	85
LETTER	110	85
DOUBLE-LETTER	150	117
MONARCH	75	38
COMMERCIAL-10	95	41
DL	86	43
C5	90	63
A3-UNCUT	120	120
A4-UNCUT	169	84
LEDGER	170	110

**INTRAYF=\*MANUAL**

Die Papierzufuhr erfolgt von Hand, daher sollten Sie jedes Mal ein Blatt Papier einlegen, wenn eine neue Seite bedruckt werden soll bzw. der Drucker Sie dazu auffordert.

Folgende Drucker können die manuelle Papierzufuhr unterstützen: 9014, 9021, 9022, 9022-200, 9026-Reno, 9026-PCL, 4812, DJET, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 9000-PCL.

**INTRAYF=\*A3**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das A3-Bögen enthält. Der Wert \*A3 ist für die Drucker 4830-PCL, 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*A4**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das A4-Bögen enthält. Der Wert \*A4 ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.



**INTRAYF=\*A5**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das A5-Bögen enthält. Der Wert \*A5 ist für die Drucker 4830-PCL, 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*B4**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das B4-Bögen enthält. Der Wert \*B4 ist für die Drucker 4830-PCL und 9026-RENO anwendbar.

**INTRAYF=\*B5**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das B5-Bögen enthält. Der Wert \*B5 ist für die Drucker 4830-PCL, 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*FOLIO**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das FOLIO-Bögen enthält.

Der Wert \*FOLIO ist für den Drucker 9026-RENO anwendbar.

**INTRAYF=\*INVOICE**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das INVOICE-Bögen enthält.

Der Wert \*INVOICE ist für die Drucker 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*EXEC**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das EXEC-Bögen enthält.

Der Wert \*EXEC ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*LEGAL**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das LEGAL-Bögen enthält.

Der Wert \*LEGAL ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*LETTER**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das LETTER-Bögen enthält.

Der Wert \*LETTER ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*DOUBLE-LETTER**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das DOUBLE-LETTER-Bögen enthält.

Der Wert \*DOUBLE-LETTER ist für den Drucker 9026-RENO anwendbar.

**INTRAYF=\*MONARCH**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das MONARCH-Bögen enthält.

Der Wert \*MONARCH ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*COMMERCIAL-10**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das COMMERCIAL-10-Bögen enthält.

Der Wert \*COMMERCIAL-10 ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL und 4825-PCL anwendbar.

**INTRAYF=\*DL**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Umschläge der Größe International-DL enthält.

Der Wert \*DL ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*C5**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Umschläge der Größe International-C5 enthält.

Der Wert \*C5 ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*A3\_UNCUT**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das ungeschnittene A3-Bögen enthält.

Der Wert \*A3\_UNCUT ist nur für PCL-Drucker anwendbar.

**INTRAYF=\*A4\_UNCUT**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das ungeschnittene A4-Bögen enthält.

Der Wert \*A4\_UNCUT ist nur für PCL-Drucker anwendbar.

**INTRAYF=\*LEDGER**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das ungeschnittene LEDGER-Bögen enthält.

Der Wert \*LEDGER ist nur für PCL-Drucker anwendbar.

**INTRAYF=<var: enum-of \_inray\_format\_set:1>**

Die Vereinbarung bzgl. des Papiereingabefachs wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

128	MANUAL
129	A3
130	A4
131	A5
132	B4
133	B5
134	DOUBLE-LETTER
135	EXEC
136	FOLIO
137	INVOICE
138	LEGAL
139	LETTER
140	MONARCH
141	COMMERCIAL-10
142	DL
143	C5
144	A3_UNCUT
145	A4_UNCUT
146	LEDGER

Diese Werte sind in der ASSEMBLER-, C- und COBOL-Schnittstelle definiert. In der C-Schnittstelle steht darum z.B. Folgendes:

```

/* intray_format_set                                     */
/* ENUM _intray_format_set                             */
<< Die Vereinbarung bzgl. des Eingabefachs wird >>
<< nicht explizit, sondern über folgende Konstantendefinition angegeben>>
<< (in der Parameterliste des Funktionsaufrufs kann darum der >>
<< in der Konstanten hinterlegte ganzzahlige Wert verwendet werden): >>
#define PRNTDOCintray_format_manual 128                /* intray_format = *manual */
#define PRNTDOCintray_format_a3 129                   /* intray_format = *a3 */
#define PRNTDOCintray_format_a4 130                   /* intray_format = *a4 */
#define PRNTDOCintray_format_a5 131                   /* intray_format = *a5 */
#define PRNTDOCintray_format_b4 132                   /* intray_format = *b4 */
#define PRNTDOCintray_format_b5 133                   /* intray_format = *b5 */
#define PRNTDOCintray_format_double-letter 134
/* intray_format = *double-letter*/

```

```

#define PRNTDOCinray_format_exec 135      /* intray_format = *exec      */
#define PRNTDOCinray_format_folio 136    /* intray_format = *folio     */
#define PRNTDOCinray_format_invoice 137  /* intray_format = *invoice1  */
#define PRNTDOCinray_format_legal 138    /* intray_format = *legal     */
#define PRNTDOCinray_format_letter 139   /* intray_format = *letter    */
#define PRNTDOCinray_format_monarch 140  /* intray_format = *monarch   */
#define PRNTDOCinray_format_commercial-10 141
/* intray_format = *commercial-10 */
#define PRNTDOCinray_format_d1 142      /* intray_format = *d1       */
#define PRNTDOCinray_format_c5 143      /* intray_format = *c5       */
#define PRNTDOCinray_format_a3_uncut 144 /* intray_format = *a3_uncut */
#define PRNTDOCinray_format_a4_uncut 145 /* intray_format = *a4_uncut */
#define PRNTDOCinray_format_ledger 146  /* intray_format = *ledger   */

```

**JVPASSW=**

Gibt das Kennwort an, mit dem die auftragsüberwachende Jobvariable (siehe Operand MONJV) geschützt worden ist.

**JVPASSW=*\*NONE***

Die Jobvariable ist nicht mit einem Kennwort geschützt.

**JVPASSW=<c-string 1..4 with-low>**

Kennwort, mit dem die Jobvariable geschützt worden ist.

**JVPASSW=<var: char: 4> / (<reg: char:4>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 4 Byte), die als Kennwort interpretiert wird, hinterlegt.

**LASTREC=(*begin, ident, length, position, occur*)**

Vereinbart den Enddatensatz, bis zu der die Datei verarbeitet werden soll. Der Enddatensatz kann durch eine Satznummer, eine Abschnittsnummer oder durch eine beliebige Zeichenfolge bestimmt werden.

Mit "begin" kann ein Enddatensatz, bis zu dem die Datei verarbeitet werden soll, angegeben werden.

Mit "ident" kann als Enddatensatz ein Datensatz gewählt werden, in dem die angegebene Zeichenfolge auftritt.

Mit "length" kann die Länge des Datensatzes angegeben werden.

Mit "position" wird die Position innerhalb des Datensatzes, ab der der angegebene String beginnt, angegeben.

Mit "occur" kann vereinbart werden, ab dem wievielten Auftreten des Datensatzes, der den String enthält, die Ausgabe enden soll.

***begin: \*END\_OF\_FILE / \*BY\_STRING\_ID / <integer 1..2147483647> / <var: int: 4> / (<reg: int:4>)***

Vereinbart den Enddatensatz, bis zu dem die Datei verarbeitet werden soll. Es kann hier der letzte Datensatz der Datei, die Nummer eines Datensatzes oder einer Abschnittsmarke sowie eine beliebige Zeichenfolge in einem Datensatz gewählt werden.

***begin: \*END\_OF\_FILE***

Die Ausgabe endet mit dem letzten Datensatz der Datei, auch wenn SECTION-Sätze angegeben sind.

***begin: \*BY\_STRING\_ID***

Als Enddatensatz wird ein Datensatz gewählt, in dem eine nachfolgend anzugebende Zeichenfolge auftritt.

***begin: <integer 1..2147483647>***

Der Enddatensatz wird als Ganzzahl von 1 bis 2147483647 angegeben.

***begin: <var: int: 4> / (<reg: int:4>)***

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Nummer des Enddatensatzes interpretiert wird, hinterlegt.

***ident: \*DUMMY / <c-string 1..60> / <var: char: 60> / (<reg: char:60>)***

Vereinbart, ob die Ausgabe mit einem Datensatz enden soll, in dem ein bestimmter String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

***ident: \*DUMMY***

Es wird kein String abdruckbarer oder hexadezimaler Zeichen definiert.

***ident: <c-string 1..60>***

Die Ausgabe soll mit dem Datensatz enden, in dem der angegebene String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

***ident: <var: char: 60> / (<reg: char:60>)***

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 60 Byte), die als String interpretiert wird, hinterlegt.

***length: 0 / <integer 1..60> / <var: int: 1> / (<reg: int:1>)***

Gibt die Länge des Datensatzes an.

***length: 0***

Voreinstellung: Der Datensatz hat die Länge 0 Byte.

***length: <integer 1..60>***

Die Länge des Datensatzes wird als Ganzzahl von 1 bis 60 angegeben.

**length:** <var: int: 1> / (<reg: int:1>)

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Länge des Datensatzes interpretiert wird, hinterlegt.

**position:** \*STD / <integer 1..2047> / <var: int: 2> / (<reg: int:2>)

Legt fest, ab welcher Position (ab welchem Byte nach dem Satzlängenfeld) der angegebene String im SECTION-Satz beginnt.

**position:** \*STD

Der gesuchte String beginnt standardmäßig am Anfang des Datensatzes:

- bei einer SAM-Datei: mit dem ersten Byte nach dem Satzlängenfeld
- bei einer ISAM-Datei mit KEY-POS=5: mit dem ersten Byte nach dem Schlüssel
- bei einer ISAM-Datei mit KEY-POS > 5: mit dem ersten Byte nach dem Satzlängenfeld.

**position:** <integer 1..2047>

Angabe eines ganzzahligen Wertes für die Position des Strings im SECTION-Satz.

**position:** <var: int: 2> / (<reg: int:2>)

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Position des Datensatzes interpretiert wird, hinterlegt.

**occur:** 1 / <integer 1..32767> / <var: int: 2> / (<reg: int:2>)

Legt fest, ab dem wievielten Auftreten des Datensatzes, der den LASTREC-String enthält, die Ausgabe abschließen soll.

**occur:** 1

Die Ausgabe soll mit dem ersten Auftreten des Datensatzes enden.

**occur:** <integer 1..32767>

Ganzzahliger Wert für das Auftreten des Datensatzes.

**occur:** <var: int: 2> / (<reg: int:2>)

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Anzahl des Auftretens interpretiert wird, hinterlegt.

### **LEFTMAR=**

Vereinbart, ob der Ausgabertext eingerückt werden soll. Der Operand LEFTMAR wird ignoriert, wenn zugleich CONTMOD=\*PHYSICAL angegeben ist.

### **LEFTMAR=\*STD**

Die Voreinstellung ist in der SPOOL-Parameterdatei hinterlegt und kann mit dem Kommando oder der SPSEVE-Anweisung SHOW-SPOOL-PARAMETERS abgefragt werden; die Information ist dem Ausgabefeld PRINT-CMD-DEFAULTS:...LEFT-MARGIN=... zu entnehmen.

**LEFTMAR=<integer 0..31>**

Für alle Drucker mit Ausnahme der RSO-Drucker: Der Ausgabertext soll um die angegebene Anzahl Spalten eingerückt werden.

Für alle RSO-Drucker:

Der Ausgabertext soll um die angegebene Anzahl \* 1/10 Zoll eingerückt werden.

**LEFTMAR=<var: int: 1> / (<reg: int:1>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Anzahl Spalten interpretiert wird, hinterlegt.

**LEFTOFF=\*IGNORE / <integer -255..255>**

Legt den Abstand des linken Randes der Druckseite vom linken Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der der Druckseite nicht beachtet wird.

**LEFTOFF=\*IGNORE**

Vorgabewert des Operanden, die Druckseite wird nicht in waagerechter Richtung auf dem Papier verschoben

**LEFTOFF=<integer -255..255>**

Die Druckseite wird um den angegebenen Weg in waagerechter Richtung auf dem Papier verschoben.

**LIBELEM=(*element, version, type*)**

Vereinbart den Namen, Version und Typ eines auszudruckenden Elements aus einer PLAM-Bibliothek. Ein Element ist vollständig definiert durch seinen Namen, seinen Typ und die Versionsnummer.

Die Sätze eines Elements sind bestimmten Satzarten zugeordnet. Es gibt 255 Satzarten. Es wird unterschieden zwischen Anwendersatzarten (1 bis 159) und Sondersatzarten (160 bis 255). Nur die Anwendersatzarten eines Elements können mit SPOOL ausgedruckt werden.

*element*: **\*DUMMY** / '<filename 1..80 with-wild>' / <var: char: 80> / (<reg: char:80>)

Vereinbart den Namen des Elementes.

*element*: **\*DUMMY**

Es soll kein Bibliothekselement ausgedruckt werden.

*element*: '<filename 1..80 with-wild>'

Der Name des Elementes wird als Zeichenfolge in Hochkommata angegeben.

**element:** <var: char: 80> / (<reg: char:80>)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 80 Byte), die als Name des Elementes interpretiert wird, hinterlegt.

**version:** \*HIGHEST / '<filename 1..40 with-wild>' / <var: char: 40> / (<reg: char:40>)

Versionsnummer des Elements, das ausgegeben werden soll.

**version:** \*HIGHEST

Das Element mit der alphabetisch höchsten Versionsbezeichnung soll ausgedruckt werden.

**version:** '<filename 1..40 with-wild>'

Die Versionsnummer des Elements wird als Zeichenfolge in Hochkommata angegeben.

**version:** <var: char: 40> / (<reg: char:40>)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 40 Byte), die als Versionsnummer des Elementes interpretiert wird, hinterlegt.

**type:** '<alphanum-name 1..12 with-wild>' / <var: char: 12> / (<reg: char:12>)

Typ des auszugebenden Bibliothekselements. Die Sätze eines LMS-Elements vom Typ C oder R gehören zu Sondersatzarten (160 bis 255). Darum können keine Sondersätze von solchen Elementen (C oder R) ausgedruckt werden.

**type:** '<alphanum-name 1..12 with-wild>'

Der Typ des auszugebenden Bibliothekselements wird als Zeichenfolge in Hochkommata angegeben.

**type:** <var: char: 12> / (<reg: char:12>)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 12 Byte), die als Typ des auszugebenden Bibliothekselements interpretiert wird, hinterlegt.

*Beispiel*

PRNTDOC LIBELEM=('LMSLIB-ELEM1',\*HIGHEST,'A')

Das Element mit dem Namen "LMSLIB-ELEM1" soll ausgedruckt werden. Es soll nach dem bezeichneten Element mit der alphabetisch höchsten Versionsbezeichnung gesucht werden. Das Element ist vom Typ "A".



**LINEPP=**

Legt fest, wieviel Zeilen (einschließlich Überschrift und Leerzeilen) auf eine Seite zu drucken sind.

**LINEPP=\*STD**

Wird kein Angabe gemacht, errechnet sich - unabhängig von der Angabe bei dem Operanden HEADLIN - die Zahl der Zeilen pro Druckseite nach folgender Formel:

$$\text{Zeilenanzahl} = P * Z - A - 6$$

Dabei bedeuten:

P = Papiergröße in Zoll

Z = Zeilendichte

A = Anzahl der Zeilen vor erstem Vertikaltabulator "Kanal 1"

*Drucker mit ladbarem Vorschubinformationspuffer*

- Der Vertikaltabulator "Kanal 1" legt die Zeile für den Druckbeginn fest. Standardmäßig werden 2 Leerzeilen vor Druckbeginn eingestellt, d.h. im Loop steht "Kanal 1" (CHANNEL 01) in der dritten Zeile.
- Ist der bei dem Operanden LINEPP angegebene Wert größer als die angegebene Zeilenanzahl im Loop, wird der im Loop vorgegebene Wert genommen.
- Ein hier angegebener Wert muss mindestens dreimal so groß sein wie der bei LINESP=\*SPACE\_1 / \*SPACE\_2 / \*SPACE\_3 angegebene Zeilenvorschub, wenn der Operand LINEPP zusammen mit den Operanden HEADLIN und LINESP angegeben wird.

**LINEPP=<integer 0..32767>**

Anzahl der Zeilen auf einer Seite.

**LINEPP=<var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Anzahl der Zeilen auf einer Seite interpretiert wird, hinterlegt.

**LINESP=(spacing, position)**

Bestimmt die Anzahl der Zeilenvorschübe bzw. die Art der Steuerzeichenauswertung.

*spacing*: **\*STD** / **\*SPACE\_1** / **\*SPACE\_2** / **\*SPACE\_3** /

**\*BY-ASA-CONTROL** / **\*BY-EBCDIC-CONTROL** / **\*BY-IBM-CONTROL** / **\*NO**

Entsprechend des Operandenwertes kann Folgendes vereinbart werden:

Die Datensätze sollen mit 1 / 2 / 3 Zeilen Abstand (**\*SPACE\_n**) ausgedruckt werden.

Der Inhalt im ersten Byte der Datensätze soll als ASA-Vorschubsteuerzeichen interpretiert werden (**\*BY\_ASA\_CONTROL**).

Der Inhalt im ersten Byte der Datensätze soll als EBCDIC-Vorschubsteuerzeichen interpretiert werden (**\*BY\_EBCDIC\_CONTROL**).

Der Inhalt im ersten Byte der Datensätze soll als IBM-Vorschubsteuerzeichen interpretiert werden (**\*BY\_IBM\_CONTROL**).

Der Inhalt im ersten Byte der Datensätze soll nicht als Vorschubsteuerzeichen interpretiert werden (**\*NO**).

Der Standardwert (**\*STD**) hängt vom Modus der Steuerzeichenauswertung (Operand **CONTMOD**) ab.

*spacing*: **<var: enum-of space\_set:1>**

Die Vereinbarung bzgl. der Anzahl der Zeilenvorschübe wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (**EQUATE**) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	<b>*STD</b>
1	<b>*SPACE_1</b>
2	<b>*SPACE_2</b>
4	<b>*SPACE_3</b>
8	<b>*BY_EBCDIC_CONTROL</b>
16	<b>*BY_ASA_CONTROL</b>
32	<b>*BY_IBM_CONTROL</b>
144	<b>*NO</b>
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

*position*: **\*STD** / **<integer 1..2040>** / **<var: int: 2>** / (**<reg: int:2>**)

Nummer des Datenbytes, in dem SPOOL das Vorschubsteuerzeichen findet. Bei Datensätzen variabler Länge werden die Felder, in denen die Länge steht, nicht zu den Daten gerechnet, d.h. nicht mitgezählt.

*position*: **\*STD**

Der Standardwert hängt vom Modus der Steuerzeichenauswertung (Operand **CONTMOD**) ab.

**position: <integer 1..2040>**

Ganzzahliger Wert für die Nummer des Datenbytes.

**position: <var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Nummer des Datenbytes interpretiert wird, hinterlegt.

### **LOCKF=**

Legt fest, ob die Datei geschützt werden soll, solange sich der SPOOLOUT-Auftrag im Wartezustand befindet (TYPE 4, siehe Ausgabe des Kommandos SHOW-PRINT-JOB-STATUS). Die Datei kann in dieser Zeit nur gelesen werden.

Banddateien werden prinzipiell nicht gesperrt. Während der Verarbeitung des SPOOLOUT-Auftrages (TYPE 5, siehe Ausgabe des Kommandos SHOW-PRINT-JOB-STATUS) ist die Datei prinzipiell geschützt - unabhängig von der Angabe bei diesem Operanden.

Ein SPOOLOUT-Auftrag wird auch dann erzeugt, wenn die auszugebende Datei durch das Kommando SECURE-RESOURCE-ALLOCATION reserviert ist. Zum Zeitpunkt der Verarbeitung des SPOOLOUT-Auftrags muss die Reservierung jedoch aufgehoben sein – sonst wird der Auftrag nicht ausgeführt.

Die auszugebende Datei bleibt bis zum Ende der Session gesperrt, wenn im Makro PRNTDOC der Operand LOCKF=\*YES angegeben wird und der Auftrag wegen der Reservierung nicht ausgeführt werden kann.

### **LOCKF=\*STD**

Es gilt der bei der Systemgenerierung festgelegte Wert (NO oder YES); er ist in der SPOOL-Parameterdatei eingetragen.

### **LOCKF=\*YES**

Die Datei ist geschützt, während der SPOOLOUT-Auftrag sich im Wartezustand befindet. LOCKF=\*YES wird ignoriert, wenn zugleich im Operanden DTYPE einer der Werte \*EAM oder \*SYSTEM\_FILE angegeben ist.

Ein Makro PRNTDOC auf ein Bibliothekselement mit gleichzeitiger Angabe von LOCKF=\*YES wird abgewiesen.

Der Dateischutz durch LOCKF=\*YES bleibt auch bestehen, wenn der SPOOLOUT-Auftrag erst im nächsten Systemlauf verarbeitet wird.

### **LOCKF=\*NO**

Die Datei ist nicht geschützt, während der SPOOLOUT-Auftrag sich im Wartezustand befindet. Die Datei kann bis zum Beginn der Verarbeitung des SPOOLOUT-Auftrags gelöscht oder geändert werden.

LOCKF=\*NO wird ignoriert für temporäre Dateien.

**LOCKF=<var: enum-of lockf\_set:1>**

Die Vereinbarung bzgl. des Dateischutzes wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*STD
1	*NO
2	*YES
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**LOOP=**

Name des Loops, der in den Vorschubinformatpuffer (VFB) geladen werden soll. Der Loop-Name darf die Zeichen '\$', '&' und '@' nicht enthalten.

**LOOP=\*STD**

Die Vorschubsteuerung für den SPOOLOUT-Auftrag soll mit dem Standard-Loop des verwendeten Formulars realisiert werden.

**LOOP='<alphanum-name 1..3>'**

Name des Loops, der den Vorschub steuern soll. Die Länge des angegebenen Loops muss mit der Länge des Standard-Loops des verwendeten Formulars übereinstimmen.

Ein Loop zur Vorschubsteuerung wird benötigt für die HP- und HP90-Drucker (3351, 3353, 2090, 2140) und Drucker der Typen 3337, 3338, 3339, 3348, 3349, 3365.

Loops sind gespeichert in der Druckersteuerdatei PRFILE. Wird kein Loop angegeben, werden die impliziten Angaben bei dem Operanden FORM benutzt. Ohne Angabe der Operanden FORM oder LOOP werden Standardwerte eingesetzt.

**LOOP=<var: char: 3> / (<reg: char:3>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 3 Byte), die als Name des Loops interpretiert wird, hinterlegt.

**MACID=CPA / ' <name 1..3>'**

Bestimmt das zweite bis einschließlich vierte Zeichen der Feldnamen und Equates.

**MF=C / D / E / L / M**

Typ des Makroaufrufs. Nähere Erläuterungen sind im Handbuch „[Makroaufrufe an den Ablaufteil](#)“ zu finden.

*Hinweis*

Die Angabe von Pointer-Variablen (Operandenwert "var:pointer") ist nur mit MF=M möglich. Zusätzlich muss mit MF=D oder MF=C eine Liste generiert werden.

**MONJV=**

Gibt die Jobvariable an, in der Informationen über die Auftragsverarbeitung abgelegt werden sollen.

**MONJV=\*NONE**

Es soll keine Jobvariable mit dem Auftrag verknüpft werden.

**MONJV=\*STD**

Die Jobvariable soll den Namen der auszudruckenden Datei (ohne Katalogkennung und ohne Benutzerkennung) erhalten. Sie wird unter der Benutzerkennung und Katalogkennung des Aufrufers erzeugt. Der Aufruf PRNTDOC MONJV=\*STD wird abgewiesen, wenn

- eine Dateigeneration angegeben wird;
- eine temporäre Datei angegeben wird;
- eine OMF- oder EAM-Datei angegeben wird;
- ein PLAM-Element angegeben wird.

Der Aufruf PRNTDOC (XX,XX),MONJV=\*STD wird abgewiesen. Grund: Wenn die Jobvariable XX für den ersten Auftrag erzeugt worden ist, dann ist sie für einen zweiten Auftrag nicht mehr verfügbar.

**MONJV='<filename 1..54 with-wild>'**

Die Jobvariable soll den als Zeichenfolge angegebenen Namen erhalten. Umfasst der Auftrag mehrere Dateien, so wird an den Namen der Jobvariablen folgendes Suffix angehängt: eine fortlaufende Zahl <1..9999>, wenn gleichzeitig FAMILY=\*YES angegeben wird.

**MONJV=<var: char: 54> / (<reg: char:54>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 54 Byte), die als Name der Jobvariablen interpretiert wird, hinterlegt.

**NOTIFPAR=<var: pointer> / \*NONE**

Mit diesem Operanden kann ein spezifischer Notification-Bereich verankert werden, der im Makro SNPPRNT beschrieben ist.

**OUTFORM=**

Gibt an, ob nur im Zeichenformat oder zusätzlich sedezimal auszugeben ist.

**OUTFORM=\*CHARACTER**

Gibt nur im Zeichenformat aus. Datensätze, die die Druckzeilenlänge überschreiten, werden abgeschnitten.

**OUTFORM=\*HEXADECIMAL**

Gibt die Datensätze im Zeichenformat und in sedezimaler Darstellung aus.

*Ausgabeformat*

Am Anfang jeder Ausgabezeile steht ein 8 Byte langer Vorspann, auf ihn folgen die Daten in der Länge von 50 Byte.

Jede Ausgabezeile wird zuerst gemäß Font gedruckt und anschließend sedezimal wiederholt.

*Aufbau der Ausgabezeile*

Spalte	Inhalt
1-4	Spaltennummer, ab der die Daten des ausgegebenen Satzes beginnen
5-8	Leerzeichen
ab 9	Zeichen des Eingabesatzes gemäß Font; die einzelnen Zeichen sind durch ein Leerzeichen getrennt. In der nächsten Zeile werden die Zeichen in sedezimaler Form wiederholt.

**OUTFORM=<var: enum-of outform\_set:1>**

Die Vereinbarung bzgl. der Ausgabe der Zeichen wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*CHARACTER
2	*HEXADECIMAL
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**OUTPART=(begin, end, dimension)**

Vereinbart, dass die gesamte Eingabedatei druckaufbereitet werden, die Ausgabe sich jedoch auf eine Untermenge aller logischen Druckseiten beschränken soll.

*begin*: **\*BEGIN OF FILE** / <integer -2147483647..2147483647> /  
<var: int: 4> / (<reg: int:4>)

Ermöglicht die Angabe einer Seiten- oder Zeilennummer der Druckdatei, mit der die Ausgabe beginnen soll, was durch den Operanden *dimension* spezifiziert wird.

*begin*: **\*BEGIN OF FILE**

Standardmäßig beginnt die Ausgabe mit dem Dateianfang. Die Seiten einer Druckdatei werden wie beim Operanden LINEPP beschrieben festgelegt (soweit nicht durch Steuerzeichen ein vorzeitiger Seitenvorschub erfolgt).

***begin*: <integer -2147483647..2147483647>**

Ganzzahliger Wert für eine Seiten- oder Zeilennummer der Druckdatei. Bei einem Wert < 0 wird die Anzahl auszugebender Seiten oder Zeilen vom Dateiende her berechnet. Bei dem Operanden *DIMENSION* wird festgelegt, ob die Angabe als Seiten- oder Zeilennummer interpretiert werden soll.

***begin*: <var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Seiten- oder Zeilennummer interpretiert wird, hinterlegt.

***end*: \*END OF FILE / <integer 1..2147483647> / <var: int: 4> / (<reg: int:4>)**

Ermöglicht die Angabe einer Seiten- oder Zeilennummer der Druckdatei, mit der die Ausgabe beendet werden soll, was durch den Operanden *dimension* spezifiziert wird. Der hier angegebene Wert muss größer sein als der bei dem Operanden *begin* angegebene.

***end*: \*END OF FILE**

Standardmäßig endet die Ausgabe mit dem Dateiende. Die Seiten einer Druckdatei werden wie beim Operanden *LINEPP* beschrieben festgelegt (soweit nicht durch Steuereichen ein vorzeitiger Seitenvorschub erfolgt).

***end*: <integer 1..2147483647>**

Ganzzahliger Wert für eine Seiten- oder Zeilennummer der Druckdatei.

***end*: <var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Seiten- oder Zeilennummer interpretiert wird, hinterlegt.

***dimension*: \*PAGES / \*LINES / <var: enum-of layout\_part\_dim\_set:1>**

Legt fest, ob die bei den Operanden *begin* und *end* angegebenen Werte als Seiten- oder Zeilennummern zu interpretieren sind.

***dimension*: \*PAGES**

Die bei den Operanden *begin* und *end* angegebenen Werte sind als Seitennummern zu interpretieren.

***dimension*: \*LINES**

Die bei den Operanden *begin* und *end* angegebenen Werte sind als Zeilennummern zu interpretieren.

**dimension: <var: enum-of layout\_part\_dim\_set:1>**

Die Vereinbarung bzgl. der Dimension wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*PAGES
2	*LINES
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**OUTTRAY=\*STD / <integer 1..99> / <var: int: 1> / \*IGNORE / \*SORTER**

Legt das Papierausgabefach für LP65-Drucker, APA-Drucker sowie für die RSO-Drucker 2030-PCL, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9014, 9015, 9026-PCL, 9026-RENO, Pagestream 55 und Pagestream 75 fest. Für PCL-Drucker gilt:

Die angegebene Nummer des Ausgabefachs wird nicht gegen die Anzahl der Ausgabefächer beim Zieldrucker geprüft. Wenn Sie ein nicht vorhandenes Ausgabefach angeben, kann das zu einem Druckerfehler führen.

Der Auftrag kann nur auf Druckern mit Einzelblattverarbeitung ausgeführt werden.

**OUTTRAY=\*STD**

Die Ausgabe erfolgt in dem Papierausgabefach, das im Geräteeintrag (SPOOL-Parameterdatei) als Standard-Papierausgabefach definiert wurde.

Die Auswahl des Papierausgabefaches ist gültig für den gesamten SPOOL-OUT-Vorgang, d.h. einschließlich der Verarbeitung von Deck- und Schlussblatt.

Die Auswahl des Papierausgabefaches durch das BS2000 kann verhindert werden durch eine PCL-Datei oder über eine Eingabe an der Bedienungskonsole des Druckers. In diesen Fällen haben die Angaben bei OUTTRAY keine Auswirkungen. Auch in der PCL-Datei kann ein Papierausgabefach definiert werden.

Die Prioritätsreihenfolge bei der Auswahl eines Papierausgabefaches ist wie folgt:

1. die im PRNTDOC angegebene Nummer des Papierausgabefaches, soweit dies nicht verhindert wird durch eine PCL-Datei oder eine Eingabe an der Druckerkonsole
2. die eventuell erfolgte Auswahl über die PCL-Datei, die in PRNTDOC angegeben wurde.
3. die eventuell erfolgte Auswahl über die PCL-Datei, die in der Formulardefinition in der SPOOL-Parameterdatei festgelegt wurde.
4. die Standardwerte, die im Geräteeintrag definiert sind.

**OUTTRAY=<integer 1..99>**

Mit diesem Wert legen Sie fest, welches Papierausgabefach für den aktuellen Auftrag benutzt wird.



**OUTTRAY=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Nummer des Papierausgabefaches interpretiert wird, hinterlegt.

**OUTTRAY=\*IGNORE**

Wenn Sie diesen Wert angeben, sendet die Druckersteuerung keinen Code zur Auswahl eines Ausgabefachs an den Drucker. Dadurch können Sie das Ausgabefach in der Prologdatei festlegen.

**OUTTRAY=\*SORTER**

Mit dieser Angabe legen Sie fest, dass der Sortiermechanismus für den aktuellen Auftrag benutzt wird; und zwar mit Hilfe des Operanden SRTMODE. Der Sortiermechanismus ist zulässig für die Drucker 9026-RENO, 4818-PCL, 4822-PCL und 4825-PCL. Dieser Sortiermechanismus kann für bis zu 20 Papierausgabefächer eingesetzt werden. Er kann nicht durch die Standardwerte kontrolliert werden, die im Geräteeintrag definiert sind.

*Hinweis*

Druckerressourcen wie PROLOG-, EPILOG-, DIA-, MEMBER und Fontzeichen-Dateien werden vor der Auswahl der Papierfachausgabe gesendet.

**OVERLAY=(face, reverse)**

Legt fest, ob ein EFO-Datendia für Drucker des Typs LP65 auf Vorder- und/oder Rückseite verwendet werden soll.

Sie müssen im Druckerspeicher vorhanden sein.

Die Operanden OVERLAY, TWOSIDE und PAGECOP sind Teil ein und desselben LP65-Druckerbefehls und insofern miteinander verknüpft. Wird nur der Operand OVERLAY spezifiziert, generiert SPOOL Standardwerte für die beiden anderen Operanden. Diese Standardwerte werden von jedem in einer PCL-Datei oder an der Druckerbedienungskonsole angegebenen Wert überschrieben.

Eine Übersicht über die möglichen Kombinationen dieser drei verknüpften Funktionen finden Sie in der Beschreibung des Druckertyps LP65 im Handbuch „SPOOL (BS2000/OSD)“.

*face:* **\*STD** / **\*NONE** / **<integer 1..127>** / **<var: int: 1>** / (**<reg: int:1>**)

Identifikationsnummer des Dias, das auf der Vorderseite verwendet werden soll.

*face:* **\*STD**

Der voreingestellte Wert für die Identifikationsnummer des Dias (bzw. \*NONE) ist in der SPOOL-Parameterdatei hinterlegt.

*face:* **\*NONE**

Es soll kein Dia auf der Vorderseite verwendet werden.

*face:* **<integer 1..127>**

Ganzzahliger Wert für die Identifikationsnummer des Dias, das auf der Vorderseite verwendet werden soll.

**face:** <var: int: 1> / (<reg: int:1>)

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Identifikationsnummer interpretiert wird, hinterlegt.

**reverse:** \*STD / \*NONE / <integer 1..127> / <var: int: 1> / (<reg: int:1>)

Identifikationsnummer des Dias, das auf der Rückseite verwendet werden soll.

**reverse:** \*STD

Der voreingestellte Wert für die Identifikationsnummer des Dias (bzw. \*NONE) ist in der SPOOL-Parameterdatei hinterlegt.

**reverse:** \*NONE

Es soll kein Dia auf der Rückseite verwendet werden.

**reverse:** <integer 1..127>

Ganzzahliger Wert für die Identifikationsnummer des Dias, das auf der Rückseite verwendet werden soll.

**reverse:** <var: int: 1> / (<reg: int:1>)

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Identifikationsnummer interpretiert wird, hinterlegt.

### **PAGECOP=**

Anzahl der Seitenkopien. Im lokalen SPOOL-Betrieb nur für die Druckertypen HP, HP90, LP65, 4830-PCL und 4850-PCL.

**PAGECOP=\***STD / <integer 0..255> / <var: int: 2> // (<reg: int: 2>)

Legt fest, wie oft jede einzelne Seite aufeinander folgend zu wiederholen ist. Zusätzliche Ausdrücke der ganzen Datei können mit dem Operanden ADDCOP angefordert werden. Jeder Makro PRNTDOC mit dem Operanden PAGECOP wird zurückgewiesen, wenn bei gleichzeitiger Angabe des Operanden LINEPP die darin angegebene Zeilenzahl größer ist als die Anzahl der Zeilen im Loop, vermindert um die Anzahl der Zeilen vor der Zeile, auf der Vertikaltabulator "Kanal 1" definiert ist.

Auf einem HP- bzw. HP90-Drucker können maximal 255 Exemplare einer Seite hintereinander gedruckt werden.

PAGECOP=255 hat die gleiche Wirkung wie PAGECOP=254: Ein Original und 254 Kopien werden gedruckt.

**PAGECOP=\***STD

Für LP65-Drucker: Anzahl der Seitenkopien wie in der PCL-Datei angegeben.

Für alle übrigen Druckertypen: PAGECOP=0.

**PAGECOP=**<integer 0..255>

Ganzzahliger Wert für die Anzahl der Seitenkopien.

**PAGECOP=<var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Anzahl der Seitenkopien interpretiert wird, hinterlegt.

**PAGEPCL=**

Gibt an, welche Seiten-Definition für den Ausdruck auf LP65- oder APA-Druckern benutzt werden soll.

**PAGEPCL=\*STD**

Für APA-Drucker:

Die in der SPSLIB festgelegte Standarddefinition soll benutzt werden; siehe dazu die Tabelle der SPSLIB Standarddefinitionen im Anhang.

Für LP65-Drucker:

Nummer der PCL-Datei, mit der die Druckdatei ausgegeben werden soll. Nur die Druckdatei selbst wird mit dieser PCL-Datei ausgedruckt. Deck- und Schlussblatt werden über die PCL-Datei gesteuert, die im Formular der SPOOL-Parameterdatei definiert ist.

*Hinweis*

Eine PCL-Datei, die es unmöglich macht, eine andere PCL-Datei über ein Kanalkommando auszuwählen, darf nicht verwendet werden.

Nach dem Starten der angegebenen PCL-Datei wird der Betriebsmodus des Druckers überprüft. Ist der EXCCW-Modus aktiv und tritt während des Druckvorganges ein Fehler auf, ist der Wiederaufsetzpunkt der letzte SECTION-Satz; wurde im Makro PRNTDOC keine Angabe bei dem Operanden SECTREC gemacht, wird die Datei von Anfang an erneut ausgedruckt.

**PAGEPCL=<integer 1..50000>**

Nur für LP65-Drucker:

Ganzzahliger Wert für die Nummer der PCL-Datei, mit der die Druckdatei ausgegeben werden soll.

**PAGEPCL=<var: int: 4> / (<reg: int:4>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 4 Byte), die als Nummer der PCL-Datei interpretiert wird, hinterlegt.

**PARAM='<name 1..27>'**

Bezeichnet die Adresse der Operandenliste (nur erlaubt bei MF-Format 2 und 3). Nähere Erläuterungen sind im Handbuch „[Makroaufrufe an den Ablaufteil](#)“ zu finden.

**POOLIND=**

Nummer des Fonts aus dem Font-Pool, mit dem der SPOOLOUT-Auftrag abgearbeitet werden soll. Die Nummer des Fonts wird durch seine Stellung bei der Definition des Font-Pools bestimmt. Der angegebene Font wird verwendet, wenn POOLIND zusammen mit CONTMOD=\*PAGE\_MODE angegeben wird.

**POOLIND=0**

Es wird keine Nummer aus dem Font-Pool angegeben.

**POOLIND=<integer 0..64>**

Ganzzahliger Wert für die Nummer des Fonts aus dem Font-Pool.

**POOLIND=<var: int: 1> / (<reg: int:1>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Nummer des Fonts interpretiert wird, hinterlegt.

**POOLNAM=**

Namen des Font-Pools (nur für HP-Drucker und HP90-Drucker), der für die Ausgabe zu verwenden ist.

**POOLNAM=\*NONE**

Es wird kein Font-Pool für die Ausgabe benötigt.

**POOLNAM='<name 1..4>'**

Bezeichnet einen Font-Pool (mit maximal 64 Fonts bei einem HP-PRINTER, maximal 46 Fonts bei einem HP90-PRINTER), aus dem ein oder mehrere Fonts zur Ausgabe verwendet werden sollen. Alle Fonts dieses Pools werden bei Ausführung des SPOOLOUT-Auftrags in den Fontpuffer geladen.

**POOLNAM=<var: char: 4> / (<reg: char:4>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 4 Byte), die als Name des Font-Pools interpretiert wird, hinterlegt.

**PREFIX=S / '<name 1..1>'**

Bestimmt das erste Zeichen der Feldnamen und Equates.

**PRJCLAS=\*JOB-CLASS / <var: int: 1> / <integer 1..255>**

Nur für den SPOOL-Administrator.

Bestimmt die Auftragsklasse des SPOOLOUT-Auftrags.

**PRJCLAS=\*JOB-CLASS**

Der Vorgabewert für die Auftragsklasse wird aus dem JOINFILE entnommen.

**PRJCLAS=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Nummer der Auftragsklasse interpretiert wird, hinterlegt.

**PRJCLAS=<integer 1..255>**

Die angegebene Auftragsklasse wird dem SPOOLOUT-Auftrag zugewiesen. Der Wert 0 (Null) ist aus Kompatibilitätsgründen nicht zulässig.

**PRJNAME=**

Auftragsname für den SPOOLOUT-Auftrag.

Der Auftragsname kann aus maximal 8 Zeichen aus der Menge (A,...Z,0,...9,@, #,\$,..) gebildet werden, darf aber nicht mit einem Bindestrich beginnen oder mit einem Punkt enden. Er darf nur dann mit einem Punkt beginnen, wenn diesem ein alphabetisches Zeichen folgt; der Punkt selbst als Bestandteil des Auftragsnamens wird in diesem Fall nicht auf dem Deckblatt abgedruckt. Die Sonderzeichenfolge Punkt und Bindestrich '.'-' darf nur in Hochkommata angegeben werden.

Der Auftragsname wird auf das Deckblatt in der dritten Großdruckzeile gedruckt und erscheint auch in der Ausgabe nach dem Kommando SHOW-PRINT-JOB-STATUS. Bei fehlender Angabe wird an seiner Stelle der Auftragsname aus dem Kommando SET-LOGON-PARAMETERS eingesetzt.

**PRJNAME=\*JOB\_NAME**

Kein eigener Auftragsname für den SPOOLOUT-Auftrag.

Wurde für den kommandogebenden Auftrag ein Auftragsname (im SET-LOGON-PARAMETERS-Kommando) vergeben, wird dieser in der dritten Großdruckzeile des Deckblatts gedruckt.

**PRJNAME=<c-string 1..8 with-low>**

Auftragsname, der für den SPOOLOUT-Auftrag vergeben werden soll (und einen zusätzlich vergebenen Auftragsnamen für den kommandogebenden Auftrag überschreibt).

**PRJNAME=<var: char: 8> / (<reg: char:8>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 8 Byte), die als Auftragsname interpretiert wird, hinterlegt.

**PRJPRIO=**

Bestimmt die Dringlichkeit des SPOOLOUT-Auftrags, mit der dieser relativ zu anderen SPOOLOUT-Aufträgen gestartet wird.

**PRJPRIO=\*JOB PRIORITY**

Voreinstellung: Der SPOOLOUT-Auftrag soll die gleiche Priorität erhalten, die der kommandogebende Auftrag hat.

**PRJPRIO=<integer 30..255>**

Ganzzahliger Wert für die Priorität, die der SPOOLOUT-Auftrag erhalten soll.

Die höchste Priorität, die Sie hier vergeben können, ist im Benutzerkatalog festgelegt und kann mit dem Kommando SHOW-USER-ATTRIBUTES abgefragt werden. Bei unzulässiger Angabe wird (wie bei fehlender Angabe) für den SPOOLOUT-Auftrag die gleiche Priorität vergeben, die der kommandogebende Auftrag auch hat.

**PRJPRIO=<var: int: 1> / (<reg: int:1>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Priorität interpretiert wird, hinterlegt.

**PRNAME=**

Spezifiziert den angeforderten Zieldrucker des Druckauftrags. Angegeben werden kann ein verteilter lokaler Druckerpool, ein nicht-verteilter lokaler Druckerpool, ein RSO-Druckerpool oder ein RSO-Drucker. Mit Distributed Print Services können Sie weitere Zieldrucker angeben. Näheres entnehmen Sie bitte dem Handbuch „[Distributed Print Services \(BS2000/OSD\)](#)“.

*Bei Angabe eines Druckerpools*

Die Ausgabe erfolgt auf einen beliebigen Drucker in dem angegebenen Geräte-Pool. Der Pool muss in der SPOOL-Parameterdatei definiert sein und kann maximal 16 RSO-Geräte oder 16 lokale SPOOL-Geräte enthalten, nicht jedoch beides gleichzeitig.

Geräte-Pools werden verwaltet mit den SPSERVE-Anweisungen ADD-, MODIFY-, REMOVE-, SHOW-PRINTER-POOL, siehe Handbuch „[SPSERVE \(BS2000/OSD\)](#)“.

Der SPOOL-OUT-Auftrag wird abgewiesen, wenn

- dem im Makro PRNTDOC angeforderten Formular nicht mindestens ein Druckertyp aus dem Geräte-Pool zugewiesen ist (siehe Kommando oder SPSERVE-Anweisung SHOW-SPOOL-FORMS).
- der Operand OVERLAY angegeben ist.

Bitte berücksichtigen Sie, dass der Pool Geräte verschiedener Typen enthalten kann: Enthält die auszudruckende Datei Steuerzeichen, die nur von einem bestimmten Druckertyp interpretiert werden, sollte ein Drucker dieses Typs im Makro PRNTDOC (implizit) angegeben werden. Eine Möglichkeit besteht darin, im Makro PRNTDOC ein Formular anzugeben, das nur für den gewünschten Druckertyp definiert ist (ADD-SPOOL-FORM).

**PRNAME=\*STD**

Der SPOOL-OUT-Auftrag soll auf dem in den SPOOL-Parametern festgelegten (PRINT-CMD-DEFAULTS) Standardgerätetyp verarbeitet werden.

**PRNAME=<c-string: c-string 1..8>**

Nur für RSO.

Symbolischer Name des RSO-Geräts, auf dem der SPOOL-OUT-Auftrag verarbeitet werden soll.

**PRNAME=<var: char 1..8>**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 8 Byte), die als Name des Druckers interpretiert wird, hinterlegt.

**PRNAME = \*IPP**

Nur für RSO.

Der SPOOL-OUT-Auftrag soll auf einem IPP-Drucker ausgeführt werden, der durch seine URL adressiert wird.

**PROCADM=(user, account, passwd)**

Nur für SPOOLOUT-Aufträge unter der Benutzerkennung TSOS (OSD V1.0) bzw. unter der Benutzerkennung mit dem Privileg PRINT-SERVICE-ADM (OSD V2.0).

Ermöglicht Ihnen, festzulegen, ob der SPOOLOUT-Auftrag unter Ihrer eigenen oder einer anderen angegebenen Benutzerkennung und Abrechnungsnummer ausgeführt werden soll.

*user:* **\*SAME** / '<alphanum-name 1..8>' / <c-string 1..8> /  
<var: char: 8> / (<reg: char:8>)

Benutzerkennung, unter der der SPOOLOUT-Auftrag ausgeführt werden soll.

*user:* **\*SAME**

Der SPOOLOUT-Auftrag soll unter der eigenen Benutzerkennung ausgeführt werden.

*user:* '<alphanum-name 1..8>' / <c-string 1..8>

Gibt als Zeichenfolge in Hochkommata eine Benutzerkennung an, unter der der SPOOLOUT-Auftrag ausgeführt werden soll.

*user:* <var: char: 8> / (<reg: char:8>)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 8 Byte), die als Benutzerkennung interpretiert wird, hinterlegt.

*account:* **\*NONE** / '<alphanum-name 1..8>' / <c-string 1..8> /  
<var: char: 8> / (<reg: char:8>)

Abrechnungsnummer, unter der der SPOOLOUT-Auftrag ausgeführt werden soll.

*account:* **\*NONE**

Es wird keine Abrechnungsnummer angegeben.

*account:* '<alphanum-name 1..8>' / <c-string 1..8>

Gibt als Zeichenfolge in Hochkommata eine Abrechnungsnummer an, unter der der SPOOLOUT-Auftrag ausgeführt werden soll.

*account:* <var: char: 8> / (<reg: char:8>)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 8 Byte), die als Abrechnungsnummer interpretiert wird, hinterlegt.

*passwd:* **\*NONE** / <c-string 1..8> / <var: char: 8> / (<reg: char:8>)

Kennwort der Benutzerkennung, unter der der SPOOLOUT-Auftrag ausgeführt werden soll.

*passwd:* **\*NONE**

Es wird kein Kennwort angegeben.

*passwd:* <c-string 1..8>

Gibt als Zeichenfolge in Hochkommata das Kennwort einer Benutzerkennung an, unter der der SPOOLOUT-Auftrag ausgeführt werden soll.

**passwd:** <var: char: 8> / (<reg: char:8>)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 8 Byte), die als Kennwort interpretiert wird, hinterlegt.

**PROGRAM=(name, string)**

Legt den Gerätenamen eines virtuellen Druckers fest, von dem der SPOOLOUT-Auftrag zu verarbeiten ist.

name = Gibt den Gerätenamen an, der in der SPOOL-Parameterdatei eingetragen ist.

string = Gibt eine spezifische Zeichenkette an das Anwendungsprogramm weiter.

**name: \*ANY / <var: char: 8> / <c-string: alphanum-name 1..8, c-string 1..8>**

Legt den Gerätenamen eines virtuellen Druckers fest, von dem der SPOOLOUT-Auftrag zu verarbeiten ist.

**name: \*ANY**

Ein beliebiger virtueller Drucker darf den SPOOLOUT-Auftrag verarbeiten.

**name: <var: char: 8>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 8 Byte) hinterlegt, die als Name des virtuellen Druckers interpretiert wird.

**name: <c-string: alphanum-name 1..8, c-string 1..8>**

Der Name des virtuellen Druckers, der den SPOOLOUT-Auftrag verarbeiten soll, wird über über eine in Hochkomma eingeschlossene Zeichenfolge angegeben.

**string: \*NONE / <var: char: 8> / <c-string: alphanum-name 1..8, c-string 1..8>**

Legt Zusatzinformationen für den virtuellen Drucker fest, von dem der SPOOLOUT-Auftrag zu verarbeiten ist.

**string: \*NONE**

Keine Zusatzinformationen für den virtuellen Drucker.

**string: <var: char: 32>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 32 Byte) hinterlegt, die als Zusatzinformation für den virtuellen Drucker interpretiert wird.

**string: <c-string: alphanum-name 1..32, c-string 1..32>**

Die Zusatzinformation für den virtuellen Drucker, der den SPOOLOUT-Auftrag verarbeiten soll, wird über eine in Hochkomma eingeschlossene Zeichenfolge angegeben.

**PRTYPE=**

Gibt an, welcher Druckertyp den Druckauftrag verarbeiten soll. Nur lokale Druckertypen dürfen angegeben werden.



**PRTYPE=\*ANY**

Vom Benutzer wird kein spezieller Druckertyp angefordert. In diesem Fall bestimmt das Subsystem SPOOL automatisch die erlaubten Druckertypen, die die Benutzeranforderung bearbeiten können. Für die Ausgabe auf RSO-Drucker und Drucker in UNIX-Systemen muss \*ANY angegeben werden.

**PRTYPE=\*LP65\_PRINTER**

Der SPOOLOUT-Auftrag soll auf einem Drucker vom Typ LP65 verarbeitet werden.

**PRTYPE=\*HP\_PRINTER**

Der SPOOLOUT-Auftrag soll auf einem Drucker vom Typ HP bzw. HP90 verarbeitet werden. Die Angabe \*HP\_PRINTER schließt die Druckertypen 2090/2140/2240(HP90) mit ein. Die Steuerzeichen für Laserdrucker vom Typ HP und HP90 sind identisch; ein SPOOLOUT-Auftrag für einen HP-PRINTER kann auf einem HP90-PRINTER verarbeitet werden und umgekehrt.

Bei der Auswahl des Druckertyps kommt es darauf an, ob zugleich der Operand CONTMOD angegeben wird oder nicht:

Druckerwahl	CONTMOD=*PAGE_MODE (Voreinstellung)	CONTMOD=(*PAGE_MODE,,*HP)
<b>PRTYPE=* ANY</b>	Die Ausgabe kann auf allen Druckertypen erfolgen.	Die Ausgabe kann auf HP- bzw. HP90-Drucker erfolgen. Andere Steuerzeichen werden in die HP-/HP90-Steuerzeichen umgewandelt.
	Für HP-/HP90-Drucker gilt außerdem: Umwandlung der OVERPRINT-Funktion in die LINE-MERGE-Funktion.	
<b>PRTYPE = *HP_PRINTER</b>	Die Ausgabe kann nur auf HP- und HP90-Druckern verarbeitet werden. Die OVERPRINT-Funktion wird in die LINE-MERGE-Funktion umgewandelt.	
	Einschränkungen PROFILE: Die Datei darf nicht das Zeichen X'FF'enthalten.	Unterstützung der HP-spezifischen Steuerzeichen. Einschränkung: Die Datei darf das Zeichen X'FF' nur als Umschaltzeichen enthalten, wenn die Steuerzeichenauswertung eingeschaltet ist.

*Hinweis für HP-/HP90-Drucker*

Wenn die Angaben im PRNTDOC eine Ausgabe auf HP- bzw. HP90-Laserdrucker bestimmen, muss eine PRFILE zur Verfügung stehen. Andernfalls wird das Kommando abgewiesen. Steht in einer Installation kein HP/HP90 zur Verfügung oder nur Geräte mit ungenügendem Ausbau (z. B. zu wenige Fonts, kein Grafikspeicher für Dias, kein Seitendrehmodul), können die Aufträge nur auf Replay-Band ausgegeben werden. Die Systembetreuung kann sich mit dem Kommando SHOW-SPOOL-JOB-STATUS über solche Aufträge informieren.

**PRTYPE=\*APA\_PRINTER**

Der SPOOLOUT-Auftrag soll auf einem 2050-APA-PRINTER, einem 2090-APA-PRINTER oder einem 2090-TWIN-PRINTER verarbeitet werden.

**PRTYPE=<var: enum-of prtype\_set:1>**

Die Vereinbarung bzgl. des Druckertyps wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*ANY
1	*LP65_PRINTER
2	*APA_PRINTER
3	*HP_PRINTER
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**RECPART=(first, last)**

Vereinbart, dass von allen ausgewählten Datensätzen nur ein bestimmter Teil jedes Satzes verarbeitet werden soll.

**first: 1 / <integer 1..32767> / <var: int: 2> / (<reg: int:2>)**

Ermöglicht die Angabe einer Byte-Nummer (Satzspalte), ab der die Datensätze einer Datei auszugeben sind. (Die Bytes eines Datensatzes sind von links nach rechts- mit 1 beginnend - durchnummeriert; ISAM-Schlüssel und Steuerzeichen sind Bestandteile eines Datensatzes).

Es ist zu unterscheiden, ob im Operanden LINESP einer der Werte \*BY\_EBCDIC\_CONTROL , \*BY\_IBM\_CONTROL oder \*BY\_ASA\_CONTROL angegeben ist oder nicht.

*Ist einer dieser Werte angegeben:*

Die Ausgabe beginnt mit dem der angegebenen Byte-Nummer folgenden Datenbyte. Das Vorschubsteuerzeichen wird unabhängig von der Angabe bei *first* ausgewertet, solange der Wert für *first* kleiner ist als die Länge des Datensatzes.

Ist der angegebene Wert > der Länge des Datensatzes, wird dieser ignoriert (d.h. es wird nicht gedruckt und nicht vorgeschoben).

*Ist keiner dieser Werte angegeben:*

Die Ausgabe beginnt mit dem der angegebenen Byte-Nummer entsprechenden Datenbyte.

**first: 1**

Die Ausgabe soll mit dem ersten Byte jedes Datensatzes beginnen.

**first: <integer 1..32767>**

Ganzzahliger Wert für die Byte-Nummer (Satzspalte), ab der die Datensätze einer Datei auszugeben sind.

**first: <var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Satzspalte interpretiert wird, hinterlegt.

**last: \*STD / <integer 1..32767> / <var: int: 2> / (<reg: int:2>)**

Diese Angabe bezeichnet das letzte Byte, das von jedem Datensatz noch gedruckt werden soll.

Es ist zu unterscheiden, ob im Operanden LINESP einer der Werte \*BY\_EBCDIC\_CONTROL , \*BY\_IBM\_CONTROL oder \*BY\_ASA\_CONTROL angegeben ist oder nicht.

*Ist einer dieser Werte angegeben:*

Die Ausgabe endet mit dem der angegebenen Byte-Nummer folgenden Datenbyte (Ausnahme: Bei der Angabe FIRST-CHARACTER für eine ISAM-Datei mit KEY-POSITION=5 endet die Ausgabe mit dem der angegebenen Byte-Nummer entsprechenden Datenbyte).

*Ist keiner dieser Werte angegeben:*

Die Ausgabe endet mit dem der angegebenen Byte-Nummer entsprechenden Datenbyte.

Sind die Datensätze länger als nach der Formulardefinition erlaubt, werden sie in der Folgezeile weitergedruckt.

**last: \*STD**

Standardwert:

- Druckzeilenende (136).
- 2048 für eine EAM-Datei, SYSLST oder SYSOUT.

**last: <integer 1..32767>**

Ganzzahliger Wert für das letzte Byte, das von jedem Datensatz noch gedruckt werden soll.

**last: <var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als letztes Byte interpretiert wird, hinterlegt.

**ROT= \*NO /ROT\_0 / \*ROT\_90 / \*ROT\_180 / \*ROT\_270 / \*ROT\_0-180 / \*ROT\_180-0 / \*ROT\_90-270 / \*ROT\_270-90 / \*BY-CONTROL-CODES\* /**

**<var: enum-of\_rotation\_set: 1>**

Legt fest, ob die zu druckenden Seiten des SPOOLOUT-Auftrags gedreht auszugeben sind; wenn ja, um wieviel Grad gedreht. Für die Druckertypen 4830-PCL und 4850-PCL können nur die Werte 0, 90, 180 und 270 angegeben werden.

Die im Drucker aufgebaute Druckseite wird um eine bestimmte Gradzahl gedreht (im Uhrzeigersinn) und auf die Papiervorlage gedruckt; zum Beispiel kann hochkant eingelegtes Papier im Querformat bedruckt werden. Bei Seitendrehung um  $90^0/270^0$  wird ein gesonderter Loop benötigt (siehe Operand ROTLOOP). Die Ausgabe erfolgt bei Angabe des Operanden - außer bei ROT=\*NO - (automatisch) auf HP-Drucker und HP90-Drucker, PCL-Drucker sowie 4830-PCL- und 4850-PCL-Drucker. SHOW-SPOOL-PARAMETERS informiert darüber, ob in der aktuellen SPOOL-Konfiguration ein HP-Drucker oder ein HP90-Drucker mit Seitendrehmodul zur Verfügung steht oder nicht:

Ausgabefeld DEVICE-TYPE:..., ROT=YES/NO.

SPOOLOUT-Aufträge mit Seitendrehung werden in den Ausgaben für die Kommandos SHOW-USER-STATUS und SHOW-PRINT-JOB-STATUS JOB-IDENTIFICATION = TSN(TSN=...) angezeigt.

Der Vorschub für gedreht auszugebende Seiten wird prinzipiell über einen eigenen Loop gesteuert (ROTLOOP). Haben Sie weder einen ROTLOOP bei PRNTDOC angegeben noch ein Formular mit definiertem ROTLOOP (siehe ADD-SPOOL-FORM), übernimmt der Standard-ROTATION-LOOP R06 im Standard-Formular die Vorschubsteuerung für die gedreht auszugebenden Seiten. Deck- und Schlussblätter werden prinzipiell nicht gedreht ausgegeben.

### **ROT=\*NO**

Seitendrehung wird nicht ausgeführt. Eventuell in der Datei vorhandene Steuerzeichen für Seitendrehung werden nicht ausgewertet.

### **ROT=\*BY-CONTROL-CODES**

Steuerzeichen zur Seitendrehung in der Datei werden ausgewertet, wenn zugleich CONTMOD=\*PHYSICAL angegeben ist.

### **ROT=\*ROT\_90 / \*ROT\_180 / \*ROT\_270**

Jede Druckseite wird (im Uhrzeigersinn) um  $90^0 / 180^0 / 270^0$  gedreht ausgegeben.

In der Datei enthaltene Steuerzeichen für Seitendrehung werden nicht ausgewertet.

Für Seitendrehung um  $90^0 / 270^0$  wird ein eigener Loop benötigt. Sie müssen dabei selbst prüfen, ob die Ausgabe mit dem angegebenen Loop nicht zu Fehlern führt. Im Fehlerfall wird der Auftrag abgewiesen.

### **ROT=\*ROT\_0\_180 / \*ROT\_180\_0 / \*ROT\_90\_270 / \*ROT\_270\_90**

Die ungeraden Seiten (Zahl vor dem zweiten Bindestrich) und geraden Seiten (Angabe nach zweitem Bindestrich) sollen unterschiedlich gedreht ausgegeben werden.

**ROT=<var: enum-of rotation\_set:1>**

Die Vereinbarung bzgl. der Seitendrehung wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

213	*NO
232	*BY_CONTROL_CODES
241	*ROT_90
242	*ROT_180
243	*ROT_270
244	*ROT_0_180
245	*ROT_90_270
246	*ROT_180_0
247	*ROT_270_90
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**ROTLOOP=**

Nur für HP-Drucker und HP90-Drucker mit Seitendrehmodul.  
Loop, mit dem die Ausgabe im Querformat gesteuert werden soll.  
Der Loop-Name darf die Zeichen '\$', '&' und '@' nicht enthalten.

**ROTLOOP=\*STD**

Die Ausgabe gedreht auszugebender Seiten soll über den Standard-ROTATION-Loop des angegebenen Formulars bzw. über den Standard-ROTATION-Loop R06 des Standardformulars (wenn im Makro PRNTDOC kein Formular angegeben wurde) gesteuert werden.

**ROTLOOP=‘<alphanum-name 1..3>‘**

In Hochkommata einzuschließende Zeichenfolge für den Namen des Loops, mit dem der Vorschub für die gedreht auszugebenden Seiten gesteuert werden soll.

**ROTLOOP=<var: char: 3> / (<reg: char:3>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 3 Byte), die als Name des Loops interpretiert wird, hinterlegt.

**RSOPAR=**

Es kann die Adresse einer Parameterliste angegeben werden, in der RSO-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**RSOPAR=\*NONE**

Die Werte sollen nicht über eine Parameterliste versorgt werden.

**RSOPAR=<var: pointer> / (<reg: pointer>)**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die RSO-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die RSO-Parameterliste hinterlegt ist (A(feld) oder Angabe eines Registers).

**SECTREC=(*ident, length, position*)**

Vereinbart, ob die Datei mit Hilfe von Abschnittsmarken strukturiert werden soll. Mit "ident" kann die Zeichenfolge (Strings) in den Datensätzen, die als Abschnittsmarken fungieren sollen, mit "length" die Länge der Zeichenfolge und mit "position" die Position des angegebenen Strings im Datensatz angegeben werden.

***ident*: \*NONE / <c-string 1..60> / <var: char: 60> / (<reg: char:60>)**

Es kann vereinbart werden, ob die Datei mit Hilfe von Abschnittsmarken strukturiert werden soll.

Welche Teile der in Abschnitte unterteilten Druckdatei ausgegeben werden sollen, wird mit dem Operanden FIRSREC vereinbart.

Als Abschnittsmarken können beliebige Zeichenfolgen (Strings) in den Datensätzen verwendet werden. Diese Strings können entweder in so genannten SECTION-Sätzen, die nicht gedruckt werden, stehen oder in gewöhnlichen Datensätzen der Druckdatei (die gedruckt werden). Die Zeichenfolge kann in Form abdruckbarer Zeichen oder hexadezimaler Zeichen angegeben werden.

Das Suchen nach dem Anfang des auszudruckenden Abschnitts wird in einer eigenen "Pseudo-Controller"-Task durchgeführt. Weder die Benutzer-Task noch der Drucker werden dabei gelockt.

Ein Pseudo-Controller schreibt die Adresse des ersten Datensatzes des Abschnittes in den SPOOL-Control-Block; anschließend kann der SPOOL-OUT-Auftrag verarbeitet werden (PREPROCESSINPUT-PARTING). Wird der gewünschte Abschnitt in der Datei nicht gefunden, erscheint eine Fehlermeldung auf dem Schlussblatt (Layout des Schlussblattes siehe [Seite 611](#)).

***ident*: \*NONE**

Die Datei soll nicht mit Hilfe von Abschnittsmarken strukturiert werden.

***ident*: <c-string 1..60>**

Gibt in Hochkommata die Zeichenfolge in den Datensätzen an.

***ident*: <var: char: 60> / (<reg: char:60>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 60 Byte), die als String interpretiert wird, hinterlegt.

***length*: 0 / <integer 1..60> / <var: int: 1> / (<reg: int:1>)**

Gibt die Länge der Zeichenfolge an.

***length*: 0**

Voreinstellung: Die Zeichenfolge hat die Länge 0 Byte.

***length: <integer 1..60>***

Die Länge der Zeichenfolge wird als Ganzzahl von 1 bis 60 angegeben.

***length: <var: int: 1> / (<reg: int:1>)***

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 1 Byte), die als Länge der Zeichenfolge interpretiert wird, hinterlegt.

***position: \*STD / <integer 1..2047> / <var: int: 2> / (<reg: int:2>)***

Legt fest, ab welcher Position (ab welchem Byte nach dem Satzlängenfeld) der angegebene String im SECTION-Satz beginnt.

***position: \*STD***

Der gesuchte String beginnt standardmäßig am Anfang des Datensatzes:

- bei einer SAM-Datei: mit dem ersten Byte nach dem Satzlängenfeld
- bei einer ISAM-Datei mit KEY-POS=5: mit dem ersten Byte nach dem Schlüssel
- bei einer ISAM-Datei mit KEY-POS > 5: mit dem ersten Byte nach dem Satzlängenfeld.

***position: <integer 1..2047>***

Ganzzahliger Wert für die Position, ab der die angegebene Zeichenfolge im SECTION-Satz beginnt.

***position: <var: int: 2> / (<reg: int:2>)***

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Position interpretiert wird, hinterlegt.

**SPSPAR=**

Es kann die Adresse einer Parameterliste angegeben werden, in der SPS-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**SPSPAR=\*NONE**

Die Werte sollen nicht über eine Parameterliste versorgt werden.

**SPSPAR=<var: pointer> / (<reg: pointer>)**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die SPS-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die SPS-Parameterliste hinterlegt ist (A(feld) oder Angabe eines Registers).

**SRTMODE=\*NO / <var: enum-of \_srtmode\_se:1> / \*GROUP / \*COLLATE / \*STACKER / \*AUTOMATIC**

Mit dieser Angabe legen Sie fest, dass der Sortiermechanismus für den aktuellen Auftrag benutzt wird. Der Sortiermechanismus ist zulässig für die Drucker 9026-RENO, 4818-PCL, 4822-PCL und 4825-PCL. Er kann für bis zu 20 Papierausgabefächer eingesetzt werden. Er kann nicht durch die Standardwerte kontrolliert werden, die im Geräteeintrag definiert sind.

**SRTMODE=\*NO**

Alle Seiten des Dokuments werden von unten nach oben in den Sortierfächern abgelegt. Ausnahme 9026-RENO: Hier werden die Seiten je nach optimaler Zugänglichkeit in ein Papierfach abgelegt.

**SRTMODE=\*GROUP**

Jedes Exemplar einer bestimmten einzelnen Seite - so viele bei PAGECOP angegeben wurden - wird einzeln in einem Sortierfach abgelegt. D.h. erst die in der Reihenfolge folgende Seiten wird wieder in dasselbe Sortierfach abgelegt. Nach dem Ende des Druckauftrags enthält jedes benutzte Sortierfach eine Kopie des ausgedruckten Dokuments.

*Beispiel*

Die Datei enthält drei Seiten, und es wurde PAGECOP=2 angegeben:

Seite 3	Seite 3	
Seite 2	Seite 2	
Seite 1	Seite 1	
Fach 1	Fach 2	Fach 3

**SRTMODE=\*COLLATE**

Alle Exemplare einer Seite - so viele bei PAGECOP angegeben wurden - werden in einem Sortierfach gesammelt. Die Exemplare der folgenden Seite werden im nächsten Sortierfach abgelegt. Die Sortierfächer werden von unten nach oben benutzt.

*Beispiel*

Die Datei enthält drei Seiten, und es wurde PAGECOP=2 angegeben:

Seite 1	Seite 2	Seite 3
Seite 1	Seite 2	Seite 3
Fach 1	Fach 2	Fach 3



**SRTMODE=\*STACKER**

Nicht für den Drucker 9026 anwendbar.

Alle gedruckten Seiten werden im Massenausgabefach des Sortiermechanismus' abgelegt; und zwar bis zu einem Maximum von 500 Seiten. Dieser Modus bietet sich dafür an, wenn eine einzige Kopie von einem sehr langen Dokument ausgedruckt werden soll.

**SRTMODE=\*AUTOMATIC**

Nur für den Drucker 9026 anwendbar.

Abhängig von der in PAGECOP angeforderten Anzahl der Kopien pro Seite und der Anzahl der Sortierfächer wird das Ablageverhalten automatisch gewählt. Die bedruckten Seiten werden wie bei \*NO unsortiert abgelegt, wenn die bei PAGECOP angegebene Anzahl der Kopien gleich der Anzahl der Sortierfächer ist. Die bedruckten Seiten werden wie bei \*GROUP nach Dokumenten sortiert, wenn die bei PAGECOP angegebene Anzahl der Kopien kleiner als die Anzahl der Sortierfächer ist. Oder die bedruckten Seiten werden wie bei \*COLLATE nach Seiten sortiert in den Sortierfächern abgelegt, wenn die bei PAGECOP angegebene Anzahl der Kopien größer als die Anzahl der Sortierfächer ist.

**SRTMODE=<var: enum-of\_srtmode\_set:1>**

Die Vereinbarung bzgl. des Sortiermechanismus wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	NO
2	GROUP
3	COLLATE
4	STACKER
5	AUTOMATIC
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

Diese Werte sind in der ASSEMBLER-, C- und COBOL-Schnittstelle definiert. In der C-Schnittstelle steht darum z.B. Folgendes:

```

/* srtmode_set                                                    */
/* ENUM _srtmode_set                                             */
<< Die Vereinbarung bzgl. des Sortiermechanismus wird >>
<< nicht explizit, sondern über folgende Konstantendefinition angegeben>>
<< (in der Parameterliste des Funktionsaufrufs kann darum der >>
<< in der Konstanten hinterlegte ganzzahlige Wert verwendet werden): >>
#define PRNTDOCsrtmode_no 1          /* srtmode =*no           */
#define PRNTDOCsrtmode_group 2      /* srtmode =*group      */

```

```
#define PRNTDOCsrtmode_collate 3 /* srtmode =*collate */
#define PRNTDOCsrtmode_stacker 4 /* srtmode =*stacker */
#define PRNTDOCsrtmode_automatic 5 /* srtmode =*automatic */
```

**STPROC=**

Vereinbart den Zeitpunkt des Ausdrucks der Systemdatei vor Beendigung des Auftrags.

**STPROC=\*IMMEDIATE**

Der SPOOLOUT-Auftrag soll sofort nach Kommandoingabe erzeugt und verarbeitet werden.

**STPROC=\*NO**

Der Operand soll nicht ausgewertet werden.

**STPROC=\*AT\_FILE\_CLOSING**

Der SPOOLOUT-Auftrag soll nach dem Schließen der Systemdatei sofort verarbeitet werden.

Eine Systemdatei wird geschlossen:

- a) bei Primärzuweisung: mit Kommando LOGOFF oder CANCEL-PRINT-JOB (von einer anderen Kennung), d.h. nach Auftragsende.
- b) bei Zuordnung zu einer katalogisierten Datei:
  - mit Kommando LOGOFF oder CANCEL-PRINT-JOB (von einer anderen Kennung), d.h. nach Auftragsende
  - erneutes Kommando ASSIGN-SYSFILE (Änderung der Zuordnung) auf dieselbe Systemdatei
- c) bei Prozedurlauf nach Erreichen der Prozedurstufe 0.

**STPROC=<var: enum-of stproc\_set:1>**

Die Vereinbarung bzgl. des Zeitpunktes wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*NO
2	*AT_FILE_CLOSING
4	*IMMEDIATE
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**SYSNN=array(16): \*DUMMY / <integer 0..101> / \*SYSLST / \*SYSOUT / <var: int: 2> / (<reg: int:2>)**

Eine SYSLST- oder SYSOUT-Systemdatei soll ausgegeben werden. Maximal dürfen 16 Elemente in einer Liste angegeben werden.

**SYSNN=\*DUMMY**

Standardwert: Es soll keine Systemdatei ausgegeben werden.

**SYSNN=<integer 0..101>**

Die Ganzzahl wird folgendermaßen interpretiert:

1..99: Zahl nn, die zur Bildung des Dateinamens SYSLSTnn verwendet werden soll.

100: Die Systemdatei SYSLST ist auszudrucken.

101: Die Systemdatei SYSOUT ist auszudrucken.

**SYSNN=\*SYSLST**

Die Systemdatei SYSLST ist auszudrucken.

**SYSNN=\*SYSOUT**

Die Systemdatei SYSOUT ist auszudrucken.

**SYSNN=<var: int: 2> / (<reg: int:2>)**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Nummer der auszugebenden Systemdatei interpretiert wird, hinterlegt.

**TOPOFF=\*IGNORE / <integer -255..255>**

Legt den Abstand des oberen Randes der Druckseite vom oberen Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der der Druckseite nicht beachtet wird.

**TOPOFF=\*IGNORE**

Die Druckseite wird nicht in senkrechter Richtung auf dem Papier verschoben.

**TOPOFF=<integer -255..255>**

Die Druckseite wird um den angegebenen Weg in senkrechter Richtung auf dem Papier verschoben.

**TRANTAB=(name, file)**

Legt fest, ob zur Verarbeitung des SPOOLOUT-Auftrags eine Code-Umsetzungstabelle zu verwenden ist und aus welcher Datei diese ggf. zu entnehmen ist. Die Code-Umsetzungstabelle ist dann notwendig, wenn das Standard-Umschaltzeichen 'FF' durch ein beliebiges Zeichen ersetzt werden soll.

*name*: **\*NONE** / '<alphanumeric-name 1..8>' / '<c-string 1..8>' /  
<var: char: 8> / (<reg: char:8>)

Legt fest, ob zur Verarbeitung des SPOOLOUT-Auftrags eine Code-Umsetzungstabelle zu verwenden ist.

*name*: **\*NONE**

Es wird keine Code-Umsetzungstabelle verwendet.

**name:** '**<alphanum-name 1..8>**' / **<c-string 1..8>**

Der Name der Code-Umsetztabelle, die zur Verarbeitung des SPOOLOUT-Auftrags verwendet werden soll, wird über eine Zeichenfolge in Hochkommata angegeben.

**name:** **<var: char: 8>** / (**<reg: char:8>**)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 8 Byte), die als Name der Code-Umsetztabelle interpretiert wird, hinterlegt.

**file:** **\*SYSTEM** / **\*STD** / '**<filename 1..44 without-userid>**' / **<c-string 1..44>** / **<var: char: 44>** / (**<reg: char:44>**)

Legt fest, aus welcher Datei die Code-Umsetztabelle zu entnehmen ist.

**file:** **\*SYSTEM**

Die angegebene Code-Umsetztabelle wird standardmäßig der Datei \$SYSSPOOL.PRFILE entnommen; für RSO-Drucker kann auch eine Benutzer-RSOFIL angegeben werden.

**file:** **\*STD**

Für RSO-Druckaufträge wird die angegebene Code-Umsetztabelle aus der angegebenen Datei (wenn FILE=<filename..>) oder aus der Standard-Ressourcendatei \$TSOS.RSOFIL (wenn FILE=\*STD/\*SYSTEM) verwendet.

Für SPOOL-Druckaufträge ist nur FILE=\*STD relevant. Andere Werte werden ignoriert und nach \*STD umgesetzt. Folglich wird die Code-Umsetztabelle immer aus der in USER-RESOURCE-FILE angegebenen benutzereigenen Ressourcendatei bzw. aus der Standard-Ressourcendatei \$SYSSPOOL.PRFILE (wenn keine benutzereigene spezifiziert ist) genommen.

**file:** '**<filename 1..44 without-userid>**' / **<c-string 1..44>**

Der Name der Datei, aus der die angegebene Code-Umsetztabelle stammt, wird über eine Zeichenfolge in Hochkommata angegeben.

**file:** **<var: char: 44>** / (**<reg: char:44>**)

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 44 Byte), die als Name der Datei interpretiert wird, hinterlegt.

## **TRUNC=**

Legt das Verhalten für den Fall fest, dass Zeilen abgeschnitten werden.

## **TRUNC=\*STD**

Es soll die Voreinstellung aus der SPOOL-Parameterdatei gültig sein. Sie können sich diesen Wert mit dem Kommando bzw. der SPSEVE-Anweisung SHOW-SPOOL-PARAMETERS ausgeben lassen (Feld: ERROR-PR=(TRUNC=)).

**TRUNC=\*DELETE\_FILE**

Die Verarbeitung des SPOOLOUT-Auftrags wird fortgesetzt (d.h. DELF=\*YES wird ausgeführt).

Auf dem Schlussblatt wird eine entsprechende Warnung ausgedruckt.

**TRUNC=\*KEEP\_FILE**

Die Verarbeitung des SPOOLOUT-Auftrags wird fortgesetzt, jedoch die Datei anschließend nicht gelöscht.

**TRUNC=<var: enum-of truncation\_set:1>**

Die Vereinbarung bzgl. der Fortsetzung des SPOOLOUT-Auftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*STD
1	*DELETE_FILE
2	*KEEP_FILE
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

**TWOSIDE=**

Legt fest, ob auf dem LP65-Drucker, PCL-Drucker, APA-Drucker oder RSO-Drucker einseitig oder doppelseitig gedruckt werden soll.

**TWOSIDE=\*STD**

Wie in der PCL-Datei definiert, wird einseitig (SIMPLEX-Modus) oder doppelseitig (DUPLEX-Modus) gedruckt.

**TWOSIDE=\*NO**

Es wird im SIMPLEX-Modus, d.h. einseitig, gedruckt.

**TWOSIDE=\*YES**

Es wird im DUPLEX-Modus, d.h. doppelseitig, gedruckt.

Der Auftrag kann außer auf LP65-Druckern mit Einzelblattverarbeitung auf den Druckern 2030-PCL, 9026-PCL, 9026-RENO, 4822-PCL, 4825-PCL, 4824-PCL, 4830-PCL und 9000-PCL ausgeführt werden.

**TWOSIDE=\*TUMBLE**

Es wird im DUPLEX-Modus, d.h. doppelseitig, gedruckt, und die Seiten werden nicht von links nach rechts, sondern von oben nach unten umgedreht.

Der Auftrag kann außer auf LP65-Druckern mit Einzelblattverarbeitung auf den Druckern 2030-PCL, 9026-PCL, 9026-RENO, 4822-PCL, 4825-PCL, 4824-PCL, 4830-PCL und 9000-PCL ausgeführt werden.

**TWOSIDE=<var: enum-of twoside\_set:1>**

Die Vereinbarung bzgl. des ein- oder doppelseitigen Druckens wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*STD
1	*YES
2	*TUMBLE
4	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

*Hinweis*

Die Funktionen TWOSIDE, OVERLAY und PAGECOP sind verknüpft und Teil ein und desselben Druckerkommandos. Wird nur einer dieser Operanden spezifiziert, werden für die übrigen Standardwerte generiert.

**USERRES=**

Bezeichnet eine Benutzerdatei, die alle benötigten Ressourcen für den Ausdruck auf unterschiedlichen Druckertypen enthält: selbsterstellte Loops, Fonts, Dias, Font-Pools, Code-Umsetzungstabellen und SPS Data Stream Definitionen. Es kann angegeben werden:

- eine Benutzer-PRFILE, die Loops, Fonts, Dia-Einträge (FOB) und Font-Pool-Einträge (Operand CHARSET) enthält.
- eine Benutzer-SPSLIB, die PAGEDEFS (Operand PAGEPCL), FORMDEFS, Fonts, Page-Segments, Dias und Raster-Bild-Daten enthält.
- Eine Benutzer-RSOFIL (nur für RSO-Betrieb), die Loops enthält.

Fehlt die Angabe einer Benutzer-PRFILE, -SPSLIB bzw. -RSOFIL, werden die entsprechenden Angaben der Datei \$SYSSPOOL.PRFILE, \$SYSSPOOL.SYSPRT.SPS.021 bzw. \$SYSSPOOL.RSOFIL entnommen.

**USERRES=\*DUMMY**

Benötigte Ressourcen werden der Datei \$SYSSPOOL.PRFILE, \$SYSSPOOL.SYSPRT.SPS.021 oder \$TSOS.RSOFIL entnommen.

**USERRES='<filename 1..44>' / <c-string 1..44>**

Name einer Benutzer-PRFILE, -SPSLIB oder -RSOFIL, die eventuell eine Katalogkennung und eine Benutzerkennung enthält. Von SPOOL wird dieser Dateiname mit dem Suffix '.PRFILE', '.SPSLIB' oder '.RSOFIL' gebraucht.

Der String darf maximal aus 28 Zeichen (ohne Katalogkennung und Benutzerkennung) bestehen, um sicherzustellen, dass diese Benutzerdatei von jeder Kennung aus aufgerufen werden kann.

Wird der Dateiname ohne Benutzerkennung angegeben, wird die Datei erst unter der Benutzerkennung des Aufrufers, anschließend unter SYSSPOOL gesucht. Wird die Datei nicht gefunden, wird das Kommando abgewiesen.

Wird eine Benutzerkennung mitangegeben, wird nur unter dieser nach der Datei gesucht.

#### *Beispiel 1*

```
PRNTDOC DNAME=(DATEI),USERRES='$XX.XX'
```

Es wird nach der Datei \$XX.XX.PROFILE gesucht. Wird die Datei nicht gefunden, so wird das Kommando abgewiesen.

#### *Beispiel 2*

Ist die Katalogkennung angegeben, wird die Suche auf den angegebenen Pubset beschränkt:

```
PRNTDOC DNAME=(DATEI), USERRES='A:XXXXX'
```

Die Datei :A:\$userid.XXXXX.PROFILE wird gesucht. Wird die entsprechende Datei nicht gefunden, wird weitergesucht nach der Datei :A:\$SYSSPOOL.XXXXX.PROFILE. Wird diese Datei nicht gefunden, wird das Kommando abgewiesen.

#### *Beispiel 3*

```
PRNTDOC DNAME=(DATEI), USERRES='A:$XX.XXXXX'
```

Die Datei :A:\$XX.XXXXX.PROFILE wird gesucht. Wird die Datei nicht gefunden, wird das Kommando abgewiesen.

Befindet sich die Datei auf einem PVS, der exportiert wurde (Kommando EXPORT-PUBSET), werden alle SPOOLOUT-Aufträge, die ihn benötigen, in die KEEP-Warteschlange gesetzt. Ist der PVS wieder verfügbar (Kommando IMPORT-PUBSET), werden die SPOOLOUT-Aufträge erneut gestartet.

#### **USERRES=<var: char: 44> / (<reg: char:44>)**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 44 Byte), die als Name der Ressourcen-Datei interpretiert wird, hinterlegt.

#### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

#### **VIRTUAL=\*STD / \*ALLOWED / <var: enum-of virtual\_set:1> / \*NOT-ALLOWED**

Dieser Operand erlaubt es, den Druckauftrag über einen virtuellen Drucker an eine Anwendung zum Bearbeiten weiterzugeben.

**VIRTUAL=\*ALLOWED**

Der Druckauftrag wird bevorzugt an ein virtuelles Gerät übergeben. Sollte beim Eintragen des Druckauftrags in die Liste der Druckaufträge kein virtuelles Gerät aktiv sein, wird der Druckauftrag einem realen Gerät zugewiesen.

**VIRTUAL=\*NOT-ALLOWED**

Dieser Druckauftrag soll nicht über ein virtuelles Gerät an eine Anwendungsprogramm weitergegeben werden.

**VIRTUAL=<var: enum-of virtual\_set:1>**

Die Vereinbarung, ob und wie ein Druckauftrag über einen virtuellen Drucker an eine Anwendung zum Bearbeiten weitergegeben wird, wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	ALLOWED
1	NOT-ALLOWED
siehe Beschreibung der Parameterliste ab <a href="#">Seite 119</a> .	

Diese Werte sind in der ASSEMBLER-, C- und COBOL-Schnittstelle definiert. In der C-Schnittstelle steht darum z.B. Folgendes:

```

/* virtual_set                                     */
/* ENUM _virtual_set                               */
<< Die Vereinbarung, ob und wie ein Druckauftrag über einen virtuellen >>
<< Drucker an eine Anwendung zum Bearbeiten weitergegeben wird, wird >>
<< nicht explizit, sondern über folgende Konstantendefinition angegeben>>
<< (in der Parameterliste des Funktionsaufrufs kann darum der >>
<< in der Konstanten hinterlegte ganzzahlige Wert verwendet werden): >>
#define PRNTDOCvirtual_allowed 0                 /* virtual = *allowed      */
#define PRNTDOCvirtual_not-allowed 1           /* virtual = *not-allowed  */

```

**VIRTUAL=\*MUST**

Der Druckauftrag wird an einen bestimmten virtuellen Drucker übergeben, der mit den Parametern in der Klammer festgelegt wird. Sollte die Supervisor-Task dieses virtuellen Druckers inaktiv sein, bleibt der Druckauftrag im Wartezustand.



**SCHEDTIME = (*schdate*, *schttime*)**

Bestimmt die Scheduling-Zeit, d.h. die Zeit ab der ein Druckauftrag frühestens gestartet wird. Damit wird der Druckauftrag erst ab dem angegebenen Zeitpunkt entsprechend der Druckerverfügbarkeit und den Abarbeitungsregeln für Druckaufträge gestartet. Im Fall, dass sich Aufträge im WFT-Status befinden, kann der Datentransfer ab diesem Moment ausgelöst werden. Es ist nicht erlaubt, diesen Operanden in Interoperabilität mit UNIX-Systemen zu bringen.

***schdate*: \*TODAY / <var: char 1..10> / <c-string 10..10>**

Bestimmt den Tag, an dem ein Druckauftrag gestartet wird.

***schdate*: \*TODAY**

Der Tag ist der gegenwärtige Tag.

***schdate*: <var: char 1..10> / <c-string 10..10>**

Bestimmt das Datum in der Form <Jahr>-<Monat>-<Tag>. Diese Angabe setzt sich aus drei unsigned Integerwerten zusammen, die durch Bindestriche getrennt werden. Die Jahresangabe hat vier Stellen. Die Monatsangabe hat bis zu zwei Stellen mit den möglichen Zahlen 1..12. Die Tagesangabe hat auch bis zu zwei Stellen mit den möglichen Zahlen 1..31, wobei die Tagesangabe mit der Monatsangabe stimmig sein muss. D.h. es darf z.B. nicht 2000-02-30 angegeben werden.

***schttime*: \*NOW / <var: char 1..5> / <c-string 5..5>**

Bestimmt den Zeitpunkt, an dem ein Druckauftrag gestartet wird.

***schttime*: \*NOW**

Der Zeitpunkt ist der gegenwärtige Zeitpunkt.

***schttime*: <var: char 1..5> / <c-string 5..5>**

Bestimmt den Zeitpunkt des Tages in der Form <Stunde>:<Minute>. Diese Angabe setzt sich aus zwei unsigned Integerwerten mit jeweils 2 Stellen zusammen, die durch einen Doppelpunkt getrennt sind. Die möglichen Zahlen sind 0..23 bei den Stunden und 0..59 bei den Minuten. Bei einstelligen Werten kann eine Null vorangestellt werden.

*Hinweise*

1. Das Datum und die Zeit, die in diesem Kommando angegeben werden, werden in der lokalen Zeitzonendarstellung ausgedrückt.
2. Um die Koheränz zwischen Client- und Serverzeit zu sichern, wird das Datum und die Zeit intern in das Format UTC (Universal Time Coordinated) konvertiert. Eine Zurückkonvertierung findet darauf jedes Mal statt, wenn Datum und Zeitangaben ausgegeben werden.

3. Das Datum und die Zeit, die durch das Kommando SHOW-PRINT-JOB-ATTRIBUTES ausgegeben werden, werden stets gemäß der lokalen Zeitdarstellung des Rechners ausgeführt, auf dem das Show-Kommando eingegeben wurde. Das bedeutet, dass die angezeigten Datum- und Zeitwerte sich von den original im PRINT-DOCUMENT-Kommando spezifizierten Angaben unterscheiden können.  
Folgende Fälle können solche Unterschiede auslösen:
  - Die Anzeige von verteilten Druckaufträgen erfolgt auf Client und Server von verschiedenen Zeitzonen.
  - Die Anzeige von verteilten Druckaufträgen erfolgt zwar auf einem Client und einem Server in der selben Zeitzone, aber ihre Uhren sind nicht synchronisiert.
  - Die Druckaufträge sind während der Sommer oder Winterzeit aufgegeben worden, die Anzeige erfolgt aber in einer anderen Zeit.
4. In einer verteilten Umgebung wird vorgeschlagen, alle Uhren zu synchronisieren, um Inkoherenzen zwischen Datums- und Zeitanzeigen zu vermeiden.
5. Wenn in einer verteilten Umgebung der SCHEDULING-TIME-Operand von der SPOOL-Version nicht unterstützt wird, wird die Angabe dieses Operanden ignoriert und der Druckauftrag wird in normaler Weise abgearbeitet.
6. Damit einem Druckauftrag ein Replay-Band zugewiesen wird, muss der Druckauftrag sein Scheduling-Time-Limit erreicht haben.

*Beispiel*

```
SCHEDTIME = (SCHDATE=*TODAY,SCHTIME=*NOW)
```

Es gibt keine Scheduling-Zeit. Der Druckauftrag wird entsprechend der Druckerverfügbarkeit und den Abarbeitungsregeln für Druckaufträge gestartet.

## Unverträgliche Operanden im Makro PRNTDOC

Der Makro PRNTDOC wird zurückgewiesen, wenn einer der Operanden in der linken Spalte gleichzeitig mit einem rechts daneben stehenden Operanden angegeben wird.

Operand	Unverträgliche Operanden
CHARSET =chars-name	PRTYPE=*LP65_PRINTER / *APA_PRINTER, CONTMOD=*APA
POOLNAM=nam	CONTMOD=*LOGICAL / *APA / *LINE-MODE / *PHYSICAL, PRTYPE=*LP65_PRINTER / *APA_PRINTER
CONTMOD=*APA	RECPART, DOCUMENT_PART (wenn keine Abschnitte angegeben), LEFTMAR, CHECKPOINT=*ON_SECTION_RECORDS
Nur für RSO: CONTMOD =*LINE_MODE / *PHYSICAL	OUTFORM=*HEXADECIMAL, ROT=*BY_CONTROL_CODES / *ROT_n CONTMOD=(..., *NO,...) OVERLAY, ROTLOOP, POOLNAM, FOB, PRTYPE=*LP65_PRINTER / *APA_PRINTER / *HP_PRINTER,
Nur für RSO: CONTMOD =*LOGICAL	OUTFORM=*HEXADECIMAL, ROT=*BY_CONTROL_CODES / *ROT_n ROTLOOP, POOLNAM, FOB, PRTYPE=*LP65_PRINTER / *APA_PRINTER / *HP_PRINTER,
CONTMOD =*PAGE-MODE(..., ..., *HP)	PRTYPE=*LP65_PRINTER / *APA_PRINTER, INTRAY OUTTRAY
DELF=*YES	DTYPE=*EAM
FAMILY=*YES	SECTREC
DTYPE=*EAM	DELF=*DESTROY
DTYPE =*LIBRARY_ELEMENT	LOCKF=*YES
EFO	PRTYPE=*APA_PRINTER, CONTMOD=*LINE_MODE / *APA / *PHYSICAL, DOCFORM=*SPECIAL_FORMAT


(Teil 1 von 2)

Operand	Unverträgliche Operanden
FOB	CONTMOD=*LOGICAL / *APA / *LINE_MODE / *PHYSICAL, DOCFORM=*SPECIAL_FORMAT, PRTYPE=*LP65_PRINTER / *APA_PRINTER
LEFTMAR	CONTMOD=*APA
LOCKF=*YES	DTYPE=*LIBRARY_ELEMENT
LOOP	PRTYPE=*APA_PRINTER, CONTMOD=*APA
OUTFORM =*HEXADECIMAL	CONTMOD=*PHYSICAL
PAGECOP	CONTMOD=*APA, PRTYPE=*APA_PRINTER
PRTYPE=*APA	CONTMOD=*LOGICAL / *PHYSICAL / *LINE_MODE / *PAGE_MODE, DOCFORM=*SPECIAL_FORMAT, PAGECOP, ROTLOOP, LOOP, ROT, CHARSET, EFO, FOB
RECPART	CONTMOD=*APA
ROT =*BY_CONTROL_CODES / =*ROT_n	CONTMOD=*LOGICAL / *APA / *LINE_MODE / *PHYSICAL DOCFORM=*SPECIAL_FORMAT / *TEXT, PRTYPE=*LP65_PRINTER / *APA_PRINTER
USERRES	PRTYPE=*LP65_PRINTER

(Teil 2 von 2)

## Returncodes

Der **PRNTDOC**-Makroaufruf wird mit einer Fehlermeldung abgewiesen, wenn eine Datei, auf die sich der Aufruf bezieht, bereits im Ausgabemodus eröffnet ist. Die nachfolgenden Returncodes werden im Makro SCPRC definiert, das als Submakro zusammen mit dem Makro PRNTDOC erzeugt wird.

R15:  Über die Ausführung des Makros PRNTDOC wird im rechtsbündigen Byte des Registers R15 ein Returncode übergeben.

Ist SPOOL nicht geladen, enthält R15 X'01000000'.

X 'aa'	Erläuterung
X '00'	Erfolgreiche Verarbeitung
X '01'	Ungültiger Dateiname
X '02'	Ungültiger PL-Zugriff
X '04'	Fehler in der Warteschlangenroutine
X '08'	Konflikt zwischen den Operanden
X '0C'	Fehler beim Durchsuchen des Katalogs
X '10'	Fehler bei der EAM-Verarbeitung
X '18'	Kennwort falsch oder nicht angegeben
X '1C'	Fehler bei \$REQM
X '20'	Fehler bei \$RELM
X '24'	Ungültige Operandenangabe
X '2C'	Kein Fremdzugriff zur Datei erlaubt
X '30'	RECFORM=F und RECSIZE=0
X '34'	Erstellungs-/Freigabedatum nicht vorhanden Für die Datei ist LASTPG=0000
X '40'	EAM-Datei nicht vorhanden
X '44'	System-Datei enthält keine Daten
X '48'	*SYSLST, *SYSLSTnn und *SYSOUT nicht zusammen mit dem Operanden DELF angeben
X '54'	Für die auszudruckende Datei ist ein DELETE-FILE-Kommando aktiv
X '6C'	FLUSH/CLOSE-Fehler bei der Systemdatei
X '70'	Datei wird gerade verwendet
X '78'	Der Aufruf PRNTDOC *SYSOUT ist in einem Dialogauftrag nicht erlaubt

(Teil 1 von 2)

X 'aa'	Erläuterung
X '7C'	Die Angabe LINESP=( <i>*BY_EBCDIC_CONTROL,..</i> ) ist für eine Datei mit RECFORM=U nicht erlaubt
X '80'	Dateigeneration nicht erlaubt
X '88'	Angesehener Zieldrucker existiert nicht
X '90'	Operanden für HP-/HP90-Drucker falsch oder nicht vorhanden
X '94'	Subsystem SPOOL nicht mehr aktiv
X '98'	Auszudruckende Systemdatei ist einer Dummy-Datei ( <i>*DUMMY</i> ) zugeordnet (Kommando ASSIGN-SYSFILE)
X '9C'	Auszudruckende Systemdatei ist einer Banddatei zugeordnet
X 'A0'	Auszudruckende Systemdatei ist einer temporären (Benutzer-) Datei zugeordnet (Kommando ASSIGN-SYSFILE)
X 'A4'	Die Datei SYSLSTnn ist nicht einer SAM-Datei zugeordnet (Kommando ASSIGN-SYSFILE)
X 'A8'	Die angegebene Datei ist für (andere) Zugriffe gesperrt
X 'B4'	Nicht existierendes Formular
X 'B8'	SAM-Fehler beim Konvertieren der EAM-Datei in eine SAM-Datei
X 'BC'	Der Operand ROT wurde angegeben ( <i>≠ *NO</i> ), obwohl die Funktion nicht zugelassen ist. (Die Systembetreuung kann dies mit M-S-P ändern.)
X 'C0'	Die Anzahl der Zeichensätze oder die Größe des gewünschten Dias übersteigt den voreingestellten Wert. (Die Systembetreuung kann die Werte mit M-S-P ändern.)
X 'C4'	Der für den Zeichensatz-Pool angegebene Index ist größer als die Anzahl der Zeichensätze des Pools.
X 'C8'	Das Seitendrehmodul steht zur Verfügung, aber die Definition für einen ROTLOOP fehlt im angegebenen Formular.
X 'CC'	Eine neue TSN kann nicht zugewiesen werden;
X 'D4'	Keine Information über Benutzerauftrag; Systemfehler.
X 'DC'	Die angegebene Bibliothek ist nicht vom Typ PLAM
X 'E0'	Die angegebene Bibliothek ist nicht verfügbar
X 'E4'	Das angegebene Element wurde nicht gefunden
X 'E8'	MONJV-Fehler
X 'EC'	Fehler in der Zulassung zur Verarbeitung
X 'F0'	Verteilte Aufträge nicht zulässig
X 'FC'	Interner Fehler
X 'FF'	Version der Schnittstelle wird nicht unterstützt

(Teil 2 von 2)

## Datenbeschreibung und Beispiel für die Programmiersprache Assembler

Der Makro PRNTDOC kann in Assembler entsprechend der Beschreibung des Formats mit direkter Angabe aller aufgeführten Operanden oder durch Angabe einer Parameterliste, in der die Operandenwerte hinterlegt sind, aufgerufen werden. Das nachfolgende Beispiel, das sich der Feldbeschreibung des Makros anschließt, zeigt jeweils die möglichen Aufrufe des Makros.

### Beschreibung der Parameterliste

\* parameter list description

SCPAPL	DS	0F	
SCPAHDR	FHDR	MF=(C,SCPA),EQUATES=NO	Standard header
* main return codes			
SCPAOK	EQU	0	successfull processing
SCPAW048	EQU	72	warning
SCPAE001	EQU	1	invalid filename
SCPAE002	EQU	2	invalid pl access
SCPAE008	EQU	8	inconsistency between param
SCPAE024	EQU	36	invalid parameter spec
SCPAE030	EQU	48	recform=F not allowed with
*			
			recsize=0
SCPAE078	EQU	120	print *sysout not allowed
SCPAE07C	EQU	124	l-spac=ebcdic and recform U
SCPAE080	EQU	128	fgg not allowed
SCPAE090	EQU	144	invalid or missing parameters
*			
			for HP-HP90
SCPAE098	EQU	152	system file assigned to
*			
			*dummy
SCPAE09C	EQU	156	system file assigned to tape
SCPAE0A0	EQU	160	system file assigned to
*			
			temporary one
SCPAE0A4	EQU	164	system file not assigned to
*			
			SAM file
SCPAE0DC	EQU	220	not a plam library
SCPAWVER	EQU	65535	interface version not
*			
			supported
SCPAE01C	EQU	28	memory request error
SCPAE020	EQU	32	memory release error
SCPAE004	EQU	4	queue routine error
SCPAE0D4	EQU	212	no information on user job
SCPAE0FC	EQU	252	internal error
SCPAE00C	EQU	12	DMS error
SCPAE010	EQU	16	error in EAM processing
SCPAE018	EQU	24	invalid or missing password
SCPAE02C	EQU	44	invalid file access
SCPAE034	EQU	52	creation-expiration date not

*			available
SCPAE040	EQU	64	EAM file not available
SCPAE044	EQU	68	syst file contains no data
SCPAE054	EQU	84	DELETE-FILE command active
SCPAE06C	EQU	108	syst file close error
SCPAE070	EQU	112	file currently in use
SCPAE088	EQU	136	destination does not exist
SCPAE0A8	EQU	168	file access not available
SCPAE0B4	EQU	180	form does not exist
SCPAE0B8	EQU	184	SAM error in EAM processing
SCPAE0BC	EQU	188	no rotation supported
SCPAE0C0	EQU	192	exceeding char-set-num or FOB
*			size
SCPAE0C4	EQU	196	exceeding char-set-num in
*			pool
SCPAE0C8	EQU	200	no rotation-loop defined
SCPAE0CC	EQU	204	new TSN can not be assigned
SCPAE0E0	EQU	224	PLAM library not available
SCPAE0E4	EQU	228	library element not found
SCPAE0E8	EQU	232	monjv error
SCPAE0EC	EQU	236	authorization error
SCPAE0F0	EQU	240	distribution not allowed
SCPAE094	EQU	148	SPOOL subsystem pended
SCPAFFFF	EQU	65535	S&P subsystems not available
*			
SCPARSOP	DS	A	address of rso_pl
SCPASPSP	DS	A	address of sps_pl
SCPADPP	DS	A	address of dprint_pl
SCPASEMP	DS	A	address of dsem_pl
SCPARES2	DS	A	free space
SCPARES3	DS	A	free space
SCPARES4	DS	A	free space
SCPADTY	DS	H	type of the document to print
*			
SCPADTYF	EQU	1	print a file
SCPADTYL	EQU	2	print a library-element
SCPADTYP	EQU	3	print a POSIX-file
SCPADTYE	EQU	4	print a EAM-file
SCPADTYS	EQU	5	print a system file
*			
SCPADNAL	DS	H	Length of the name of the
*			document(s) to be printed
SCPADNAP	DS	A	Address of the name of the
*			document(s) to be printed
SCPAEAMN	DS	16F	EAM-file-number(s) to be
*			printed
SCPASYSN	DS	16H	*SYSOUT/SYSLSTnn to be
*			printed
*			
SCPALIBE	DS	0XL132	Library element(s) to be



*			printed
SCPAELNA	DS	CL80	element name
SCPAELVE	DS	CL40	element version
SCPAELTY	DS	CL12	element type
*			
*			
SCPASECR	DS	0XL64	Identifier of the optional
*			section records
SCPASEID	DS	CL60	section id
SCPASEPO	DS	H	section id position
SCPASELE	DS	X	section id length
SCPASEIN	DS	AL1	sect_ind
SCPASEGI	EQU	X'80'	This bit must be set if
*			sect-id is given, i.e. if it
*			is not *NONE.
SCPASEIC	EQU	X'40'	section-id is given in
*			character (= true) or
*			hexadecimal (= false). This
*			field serves to indicate in
*			which format the section-id
*			should be printed on the
*			trailer-page.
SCPASERF	EQU	X'3F'	not used
*			
*			
SCPAFREC	DS	0XL72	Beginning part of the
*			document(s) to be printed
SCPASPOS	DS	F	record number
SCPASSID	DS	CL60	string-id
SCPASSPO	DS	H	string-id position
SCPASTOC	DS	H	string-id occurrence
SCPASSLE	DS	X	string-id length
SCPAFIL6	DS	CL1	not used
SCPASSII	DS	Y	strid_ind
SCPASSIC	EQU	X'8000'	string-id is given in
*			character (= true) or
*			hexadecimal (= false). This
*			field serves to indicate in
*			which format the section-id
*			should be printed on the
*			trailer-page.
SCPASSIF	EQU	X'7FFF'	not used
*			
*			
SCPALREC	DS	0XL72	End part of the document(s)
*			to be printed
SCPAEPOS	DS	F	last record number
SCPAESID	DS	CL60	string-id
SCPAESPO	DS	H	string-id position
SCPAETOC	DS	H	string-id occurrence

SCPAESLE	DS	X	string-id length
SCPAFIL7	DS	CL1	not used
SCPAESII	DS	Y	strid_ind
SCPAEEIC	EQU	X'8000'	string-id is given in
*			character (= true) or
*			hexadecimal (= false). This
*			field serves to indicate in
*			which format the section-id
*			should be printed on the
*			trailer-page.
SCPAESIF	EQU	X'7FFF'	not used
*			
*			
SCPARECP	DS	0XL4	Part of the input record to
*			be processed
SCPARPFC	DS	H	first character
SCPARPLC	DS	H	last character
*			
*			
SCPALISP	DS	0XL4	Number of line feeds after a
*			line has been printed
SCPACCP0	DS	H	control char position
SCPASPAC	DS	FL1	line spacing
*			
* _space_set			
SCPASPA1	EQU	1	spacing = *space_1
SCPASPA2	EQU	2	spacing = *space_2
SCPASPA3	EQU	4	spacing = *space_3
SCPASPAE	EQU	8	spacing = *by_ebcdic_control
SCPASPAA	EQU	16	spacing = *by_asa_control
SCPASPAI	EQU	32	spacing = *by_ibm_control
SCPASPAN	EQU	144	spacing = *no
SCPASPAS	EQU	0	spacing = *std
*			
SCPAFIL4	DS	CL1	not used
*			
SCPADOCF	DS	FL1	document format
*			
* _docform_set			
SCPADCFT	EQU	0	docform = *text
SCPADCFP	EQU	1	docform = *page-format
SCPADCFS	EQU	2	docform = *special-format
*			
*			
SCPACONM	DS	0XL3	Control mode applied by the
*			Spool & Print subsystem on
*			the document(s) copies
SCPACTLM	DS	FL1	control mode
*			
* _control_mode_set			
SCPACTMS	EQU	1	mode = *std
SCPACTPH	EQU	2	mode = *physical
SCPACTMP	EQU	4	mode = *page-mode

```

SCPACTML EQU 8 mode = *logical
SCPACTLI EQU 16 mode = *line-mode
SCPACTMA EQU 64 mode = *apa
*
SCPAPCC DS FL1 page control character
* _pcc_set
SCPATOPY EQU 0 pcc = *yes
SCPATOPN EQU 1 pcc = *no
SCPATOPD EQU 2 pcc = *dummy
*
SCPACTLT DS FL1 control type
* _control_type_set
SCPACTDU EQU 0 ctltype = *dummy
SCPACTHC EQU 1 ctltype = *compatible
SCPACTHP EQU 2 ctltype = *hp
*
*
SCPAFAM DS FL1 a common TSN is allocated if
* a number of files or library
* elements are specified
* _family_set
SCPAFAMS EQU 1 family = *std
SCPAFAMY EQU 2 family = *yes
SCPAFAMN EQU 3 family = *no
*
*
SCPACHKP DS FL1 Checkpoint processing is to
* be performed by the
* controller on the basis of
* pages or section
* _checkp_set
SCPACHPP EQU 0 checkp = *on-page
SCPACHPR EQU 1 checkp = *on-section-record
*
*
SCPADOCN DS H Number of files
SCPASPV DS F ONLY TPR: Time of printout
* for the system file prior to
* termination of the job
*
SCPASPS DS FL1 ONLY TPR: specifies whether
* time of printout for the
* system file is prior to
* termination of the job
* _stproc_set
SCPASTPV EQU 0 stproc = value TPR only
SCPASTPN EQU 1 stproc = no TPR only
SCPASTPC EQU 2 stproc = close TPR only
SCPASTPI EQU 4 stproc = immediate TPR only
*
*
SCPALOCK DS FL1 Lock while job is waiting
* _lockf_set
SCPALOCS EQU 0 lockf = *std

```

SCPALOCN	EQU	1	lockf = *no
SCPALOCY	EQU	2	lockf = *yes
*			
SCPADEL	DS	FL1	Delete (destroy) document
*			
_delf_set			
SCPADELY	EQU	1	delf = *yes
SCPADELD	EQU	2	delf = *destroy
SCPADELN	EQU	3	delf = *no
*			
SCPACOPY	DS	X	how many additional times the
*			print-job must be repeated
*			(number of additional
*			document(s) copies)
SCPAJVP	DS	0XL4	jvpass
SCPAJVPI	DS	F	Password with which the job
*			variable is protected
*			(numeric)
	ORG	SCPAJVP	
SCPJVPI	DS	CL4	Password with which the job
*			variable is protected
	ORG	SCPAJVP+4	
SCPMJV	DS	CL54	Job variable into which
*			information on print-job
*			processing is to be stored
SCPAPRJN	DS	CL8	Job-name of the print-job
SCPAPRIO	DS	X	Priority of the print-job
*			
SCPAPRAD	DS	0XL24	Allows to execute the
*			print-job under another
*			user-id
SCPAPRAU	DS	CL8	user-id under which the
*			print-job is launched
SCPAPRAA	DS	CL8	account number under which
*			the print-job is launched
SCPAPRAP	DS	CL8	password for the user-id
*			given with the <user>
*			parameter
*			
SCPATRNC	DS	FL1	Line truncation
*			
_truncation_set			
SCPATRUS	EQU	0	trunc = *std
SCPATRUD	EQU	1	trunc = *delete_file
SCPATRUK	EQU	2	trunc = *keep_file
*			
*			
SCPAPART	DS	0XL12	layout control (number of
*			pages or lines)
SCPAPRCFR	DS	F	from record
SCPAPRCTO	DS	F	till record
SCPALAYD	DS	FL1	dimension

```

*   _layout_part_dim_set
SCPALAYN EQU 0           dim = *none
SCPALAYP EQU 1           dim = *pages
SCPALAYL EQU 2           dim = *line
*
SCPAFIL5 DS CL3         not used
*
SCPACLAS DS X           job class
SCPALFTM DS X           Indentation for the output
*                       text
SCPALIPP DS H           Lines per page (including
*                       header and blanks)
SCPAHDLI DS FL1        Type of header line
*   _headline_set
SCPAHLNO EQU 1         headlin = *no
SCPAHLST EQU 2         headlin = *std
SCPAHLR EQU 4          headlin = *frecord
SCPAHLD EQU 8          headlin = *date
SCPAHLDR EQU 12        headlin = *date_frecord
SCPAHLP EQU 16         headlin = *page
SCPAHLRP EQU 20        headlin = *frecord_page
SCPAHLDP EQU 24        headlin = *date_page
SCPAHDRP EQU 28        headlin = *date_frecord_page
*
SCPAOUTF DS FL1        The output format is
*                       character format only or
*                       character and hexadecimal
*                       format
*   _outform_set
SCPAOUTC EQU 1         outform = *character
SCPAOUTH EQU 2         outform = *hexadecimal
*
SCPATWSD DS FL1        The document is to be printed
*                       on one or both paper sides
*                       (supported only on LP65
*                       printers)
*   _twoside_set
SCPATWOS EQU 0         twoside = *std
SCPATWOY EQU 1         twoside = *yes
SCPATWOT EQU 2         twoside = *tumble
SCPATWON EQU 4         twoside = *no
SCPATWOI EQU 8         twoside = *ignore
*
SCPAROT DS FL1        The pages to be printed from
*                       the print-job are rotated
*   _rotation_set
SCPAROTN EQU 213       rot = *no
SCPAROTO EQU 240       rot = *rot_0
SCPAROT1 EQU 241      rot = *rot_90
SCPAROT2 EQU 242      rot = *rot_180

```

```

SCPAROT3 EQU 243
SCPAROT4 EQU 244
SCPAROT5 EQU 245
SCPAROT6 EQU 246
SCPAROT7 EQU 247
SCPAROTY EQU 232
*
SCPAIT# DS 0XL1
SCPAINT DS X
*
*
ORG SCPAIT#
SCPAITS DS FL1
* _intray_set
SCPAITBF EQU 0
SCPAITIG EQU 255
*
ORG SCPAIT#
SCPAITFS DS FL1
* _intray_format_set
SCPAITMA EQU 128
SCPAITA3 EQU 129
SCPAITA4 EQU 130
SCPAITA5 EQU 131
SCPAITB4 EQU 132
SCPAITB5 EQU 133
SCPAITDB EQU 134
*
SCPAITEX EQU 135
SCPAITFO EQU 136
SCPAITIV EQU 137
SCPAITLG EQU 138
SCPAITLT EQU 139
SCPAITMO EQU 140
SCPAIT10 EQU 141
*
SCPAITDL EQU 142
SCPAITC5 EQU 143
SCPAIT3U EQU 144
SCPAIT4U EQU 145
SCPAITLD EQU 146
*
ORG SCPAIT#+1
SCPAOT# DS 0XL1
SCPAOUTT DS X
*
*
ORG SCPAOT#
SCPAOTS DS FL1
* _outtray_set
rot = *rot_270
rot = *rot_0_180
rot = *rot_90_270
rot = *rot_180_0
rot = *rot_270_90
rot = *by_control_codes
intray
Paper input tray to be used
(only on LP65 and RSO
printers)
intray = *by-format
intray = *ignore
intray_format = *manual
intray_format = *a3
intray_format = *a4
intray_format = *a5
intray_format = *b4
intray_format = *b5
intray_format =
*double-letter
intray_format = *exec
intray_format = *folio
intray_format = *invoice
intray_format = *legal
intray_format = *letter
intray_format = *monarch
intray_format =
*commercial-10
intray_format = *d1
intray_format = *c5
intray_format = *A3-uncut
intray_format = *A4-uncut
intray_format = *ledger
outtray
Paper output tray to be used
(only on LP65 and RSO
printers)

```

```

SCPAOUTI EQU 128          outtray = *ignore
SCPAOUTS EQU 129          outtray = *sorter
*
      ORG SCPAOT#+1
SCPAPGCP DS H              Copies of each printed page
*
SCPACVPG DS 0XL40         Cover-page
SCPAHTXT DS CL32          header page text
SCPAHDEX DS F             header exit number
SCPATREX DS F             trailer exit number
*
SCPAFORM DS CL6           Form-name to be used
SCPALoop DS CL3           Loop-name to be used
SCPARTLP DS CL3           Rotation-loop name to be used
SCPACHS DS 16CL3         One or more character-sets
*                          used for printing
SCPACHS# EQU 16
SCPAPLNM DS CL4           Pool-name
SCPAPLIN DS X             Pool-index
SCPAEFO DS CL2           Film or EFO overlay for
*                          printing
*
SCPAOVLY DS 0XL2         Double sided overlay (only
*                          for LP65 printer type)
SCPAOVFA DS X            face side
SCPAOVRE DS X            reverse side
*
SCPAFOB DS CL4           FOB (Forms Overlay Buffer)
*                          data overlay to be used for
*                          printing the document (only
*                          supported on HP printer
*                          equipped with a graphic
*                          buffer
SCPAVIRT DS FL1          allow virtual controller
* _virtual_set
SCPAMUST EQU 4            virtual = *must
SCPAALLO EQU 2           virtual = *allowed
SCPANOTA EQU 1           virtual = *not_allowed
SCPASTDA EQU 0           virtual = *std
*
SCPAFIL2 DS CL2          slack bytes
SCPAPGPC DS F            PCL file number (only for
*                          LP65 printer type)
SCPAUSRR DS CL44         User resources file to be
*                          used for print resources
*                          (user PRFILE generated by
*                          PRM)
*
SCPATRTB DS 0XL52        Translation table to be used
SCPATRNB DS CL8          translation table name

```

SCPATRFI	DS	CL44	translation table file
*			
SCPAPRNM	DS	CL8	Printer or printer pool name
*			on which the document is to
*			be printed
SCPAPRTY	DS	FL1	Printer type requested
*		_prtype_set	
SCPAPTAN	EQU	0	prtype = *any
SCPAPT65	EQU	1	prtype = *lp65_printer
SCPAPTAP	EQU	2	prtype = *apa_printer
SCPAPTHP	EQU	3	prtype = *hp_printer
*			
SCPASRTM	DS	FL1	Sorter mode
*		_srtmode_set	
SCPASMNO	EQU	1	srtmode = *no
SCPASMGR	EQU	2	srtmode = *group
SCPASMCL	EQU	3	srtmode = *collate
SCPASMST	EQU	4	srtmode = *stacker
SCPASMAU	EQU	5	srtmode = *automatic
*			
SCPAFIL3	DS	CL2	slack bytes
SCPATSN	DS	CL4	output: TSN obtained from the
*			system
SCPACLUS	DS	CL8	for compatibility with
*			MODIFY-PRINT-JOB
SCPASP1	DS	AL1	specified1
SCPASP11	EQU	X'80'	document type
SCPASP12	EQU	X'40'	document name
SCPASP13	EQU	X'20'	document number
SCPASP14	EQU	X'10'	library element
SCPASP15	EQU	X'08'	section record
SCPASP16	EQU	X'04'	first record
SCPASP17	EQU	X'02'	last record
SCPASP18	EQU	X'01'	record part
SCPASP2	DS	AL1	specified2
SCPASP21	EQU	X'80'	line spacing
SCPASP22	EQU	X'40'	document format
SCPASP23	EQU	X'20'	control mode
SCPASP24	EQU	X'10'	additional copies
SCPASP25	EQU	X'08'	family
SCPASP26	EQU	X'04'	checkpoint
SCPASP27	EQU	X'02'	monjv
SCPASP28	EQU	X'01'	jv-password
SCPASP3	DS	AL1	specified3
SCPASP31	EQU	X'80'	print-job-name
SCPASP32	EQU	X'40'	print-job-priority
SCPASP33	EQU	X'20'	processing admission
SCPASP34	EQU	X'10'	line-truncation
SCPASP35	EQU	X'08'	lock-file
SCPASP36	EQU	X'04'	delete-after-print



SCPASP37	EQU	X'02'	output part
SCPASP38	EQU	X'01'	page-copies
SCPASP4	DS	AL1	specified4
SCPASP41	EQU	X'80'	left-margin
SCPASP42	EQU	X'40'	line-per-page
SCPASP43	EQU	X'20'	header-line
SCPASP44	EQU	X'10'	output-format
SCPASP45	EQU	X'08'	two-sided
SCPASP46	EQU	X'04'	rotation
SCPASP47	EQU	X'02'	input-tray
SCPASP48	EQU	X'01'	output-tray
SCPASP5	DS	AL1	specified5
SCPASP51	EQU	X'80'	cover-page
SCPASP52	EQU	X'40'	form-name
SCPASP53	EQU	X'20'	loop-name
SCPASP54	EQU	X'10'	rotation-loop-name
SCPASP55	EQU	X'08'	character-set
SCPASP56	EQU	X'04'	pool-name
SCPASP57	EQU	X'02'	pool-index
SCPASP58	EQU	X'01'	electronic-form-overlay
SCPASP6	DS	AL1	specified6
SCPASP61	EQU	X'80'	overlay
SCPASP62	EQU	X'40'	fob
SCPASP63	EQU	X'20'	lp65-pagedef
SCPASP64	EQU	X'10'	user resources file
SCPASP65	EQU	X'08'	translation table
SCPASP66	EQU	X'04'	printer-name
SCPASP67	EQU	X'02'	printer-type
SCPASP68	EQU	X'01'	tsn
SCPASP7	DS	AL1	specified7
SCPASP71	EQU	X'80'	rso parameters
SCPASP72	EQU	X'40'	sps parameters
SCPASP73	EQU	X'20'	dprint parameters
SCPASP74	EQU	X'10'	dsem parameters
SCPASP7F	EQU	X'07'	
SCPASP8	DS	AL1	specified8
SCPASP81	EQU	X'80'	start-processing
SCPASP82	EQU	X'40'	sorter mode
SCPASP83	EQU	X'20'	virtual control
SCPASP84	EQU	X'10'	job-class
SCPASP85	EQU	X'08'	prog. name
SCPASP86	EQU	X'04'	string field
SCPASP87	EQU	X'02'	format name
SCPASP8F	EQU	X'01'	
*			
SCPAVALL	DS	0XL40	virtual device synchrone
SCPAPVNA	DS	CL8	virtual program name
SCPAPVST	DS	CL32	virtual program string
*			
SCPAFRMT	DS	CL63	format name

SCPARES9	DS	XL2	reserved for modify
SCPASP9	DS	AL1	specified9
SCPASP91	EQU	X'80'	top-offset
SCPASP92	EQU	X'40'	left-offset
SCPASP93	EQU	X'20'	scheduling time
SCPASP9F	EQU	X'3F'	
SCPATOP0	DS	H	Specifies the length of the
*			white band that must be
*			inserted
SCPALEF0	DS	H	Specifies the width of the
*			white band that must be
*			inserted
SCPADATE	CL	10	Specifies the date from which the
*			job can be started
SCPATIME	CL	5	Specifies the time from which the
*			job can be started
SCPAUN9	DS	XL11	unused
SCPA#	EQU	*-SCPAHDR	

*Beispiel*

```

TC5103  START
TC5103  AMODE ANY
TC5103  RMODE ANY
        GPARMOD 31
*
        PRINT NOGEN
*
TC5103  @ENTR TYP=M,LOCAL=WORKAREA,ENV=SPLSPEC
*
        @DATA CLASS=B,BASE=R9,DSECT=CONSTANT
        L      R9,=A(CONSTANT)
        L      R12,=V(ITSRVR)
        EJECT
*****
* TEST 1 : FORM RESOURCE *
*****
        MVC TESTNR,=CL30'TEST1 (FORM) OF SPGF5103'
        MVC TCDESC,TESTNR
        VTCSET LOG
        WROUT PRESENT,TERM,VTSUCBA=VTSUPAR
        WRTRD PARFRM,,VALFRM,,TERM,PARMOD=31,VTSUCBA=VTSUPAR
        LA    R4,PLC
        LA    R5,SCPA#
        LA    R6,PLL
        LR    R7,R5
        MVCL R4,R6
        PRINT GEN
    
```

```

LA    R4, FILENAMF
ST    R4, POINTER
PRNTDOC MF=M, DNAME=( POINTER, ), FORM=FORMNAME
PRNTDOC MF=E, PARAM=PLC
PRINT NOGEN
UNPK  APIRET(9), SCPARET(5)
MVI   APIRET+8, X'40'
NC    APIRET, HEXNC
TR    APIRET(8), HEXTAB
WROUT MSGRET, TERM
WRTRD CONT, , KEY, , , TERM, PARMOD=31
EJECT

```

\*

```

*****

```

```

* TEST 2 : LOOP RESOURCE *

```

```

*****

```

\*

```

MVC  TESTNR, =CL30*TEST2 (LOOP) OF SPGF5103'
MVC  TCDESC, TESTNR
WROUT PRESENT, TERM, VTSUCBA=VTSUPAR
WRTRD PARLOOP, , VALLOOP, , , TERM, PARMOD=31, VTSUCBA=VTSUPAR
LA    R4, PLC
LA    R5, SCPA#
LA    R6, PLL
LR    R7, R5
MVCL  R4, R6
PRINT GEN
LA    R4, FILENAML
ST    R4, POINTER
PRNTDOC MF=M, DNAME=( POINTER, ), LOOP=LOOPNAME
PRNTDOC MF=E, PARAM=PLC
PRINT NOGEN
UNPK  APIRET(9), SCPARET(5)
MVI   APIRET+8, X'40'
NC    APIRET, HEXNC
TR    APIRET(8), HEXTAB
WROUT MSGRET, TERM
WRTRD CONT, , KEY, , , TERM, PARMOD=31
EJECT

```

\*

```

*****

```

```

* TEST 3 : ROTATION LOOP RESOURCE *

```

```

*****

```

\*

```

MVC  TESTNR, =CL30*TEST3 (ROTLOOP) OF SPGF5103'
MVC  TCDESC, TESTNR
WROUT PRESENT, TERM, VTSUCBA=VTSUPAR
WRTRD PARROTL, , VALROTL, , , TERM, PARMOD=31, VTSUCBA=VTSUPAR

```

```

LA    R4,PLC
LA    R5,SCPA#
LA    R6,PLL
LR    R7,R5
MVCL  R4,R6
PRINT GEN
LA    R4,FILENAMR
ST    R4,POINTER
PRNTDOC MF=M,DNAME=(POINTER,),ROTLLOOP=ROTLNAME
PRNTDOC MF=E,PARAM=PLC
PRINT NOGEN
UNPK  APIRET(9),SCPARET(5)
MVI   APIRET+8,X'40'
NC    APIRET,HEXNC
TR    APIRET(8),HEXTAB
WROUT MSGRET,TERM
WRTRD CONT,,KEY,,TERM,PARMOD=31
EJECT
    
```

\*

\*\*\*\*\*

\* TEST 4 : CHARACTER SET RESOURCE \*

\*\*\*\*\*

\*

```

MVC   TESTNR,=CL30'TEST4 (CHARSET) OF SPGF5103'
MVC   TCDESC,TESTNR
WROUT PRESENT,TERM,VTSUCBA=VTSUPAR
WRTRD PARCHAR,,VALCHAR,,TERM,PARMOD=31,VTSUCBA=VTSUPAR
LA    R4,PLC
LA    R5,SCPA#
LA    R6,PLL
LR    R7,R5
MVCL  R4,R6
PRINT GEN
LA    R4,FILENAMC
ST    R4,POINTER
PRNTDOC MF=M,DNAME=(POINTER,),CHARSET=CHARNAME
PRNTDOC MF=E,PARAM=PLC
PRINT NOGEN
UNPK  APIRET(9),SCPARET(5)
MVI   APIRET+8,X'40'
NC    APIRET,HEXNC
TR    APIRET(8),HEXTAB
WROUT MSGRET,TERM
WRTRD CONT,,KEY,,TERM,PARMOD=31
    
```

\*

TERM TERM

\*

@END

```

        EJECT
WORKAREA @PAR  D=YES
*
        PRINT GEN
PLC      PRNTDOC MF=C
        PRINT NOGEN
WORKAREA @PAR  LEND=YES
*
        EJECT
POINTER DS    A
*
VTSUPAR VTSUCB MODE=EXTEND
*
PARFRM  DS    0H
        DC    Y(EPARFRM-PARFRM)
        DS    3X
        DC    AL1(LOGNL)
        DC    C'FILE NAME : ,
        DC    AL1(LOGEPA)
        DC    CL54' ,
        DC    AL1(LOGNL)
        DC    C'FORM NAME : ,
        DC    AL1(LOGEPA)
        DC    CL6' ,
        DC    AL1(LOGNL)
EPARFRM EQU    *
PARLOOP DS    0H
        DC    Y(EPARLOOP-PARLOOP)
        DS    3X
        DC    AL1(LOGNL)
        DC    C'FILE NAME : ,
        DC    AL1(LOGEPA)
        DC    CL54' ,
        DC    AL1(LOGNL)
        DC    C'LOOP NAME : ,
        DC    AL1(LOGEPA)
        DC    CL3' ,
        DC    AL1(LOGNL)
EPARLOOP EQU    *
PARROTL DS    0H
        DC    Y(EPARROTL-PARROTL)
        DS    3X
        DC    AL1(LOGNL)
        DC    C'FILE NAME           : ,
        DC    AL1(LOGEPA)
        DC    CL54' ,
        DC    AL1(LOGNL)
        DC    C'ROTATION LOOP NAME : ,

```

```

                DC    AL1(LOGEPA)
                DC    CL3' ,
                DC    AL1(LOGNL)
EPARROTL EQU   *
PARCHAR DS     0H
                DC    Y(EPARCHAR-PARCHAR)
                DS    3X
                DC    AL1(LOGNL)
                DC    C'FILE NAME           : ,
                DC    AL1(LOGEPA)
                DC    CL54' ,
                DC    AL1(LOGNL)
                DC    C'CHARACTER SET NAME : ,
                DC    AL1(LOGEPA)
                DC    CL3' ,
                DC    AL1(LOGNL)
EPARCHAR EQU   *
PRESENT DS     0H
                DC    Y(EPRESENT-PRESENT)
                DS    3X
                DC    AL1(LOGNP)
                DC    70C '**'
                DC    AL1(LOGNL)
                DC    C '**'
TCDESC DS     CL30
                DC    AL1(LOGNL)
                DC    70C '**'
                DC    AL1(LOGNL)
EPRESENT EQU   *
VALFRM DS     0CL64
                DS    CL4
FILENAMF DS    CL54
FORMNAME DS    CL6
VALLOOP DS    0CL61
                DS    CL4
FILENAML DS    CL54
LOOPNAME DS    CL3
VALROTL DS    0CL61
                DS    CL4
FILENAMR DS    CL54
ROTLNAME DS    CL3
VALCHAR DS    0CL61
                DS    CL4
FILENAMC DS    CL54
CHARNAME DS    CL3
CONT DS       0H
                DC    Y(ESCONT-CONT)
                DS    3X
    
```

```
          DC    AL1(LOGNL)
          DC    C'PLEASE ACKNOWLEDGE'
ECONT    EQU    *
KEY      DS    OCL5
          DS    CL4
          DS    CL1
*
          EJECT
CONSTANT DS    0H
*
HEXTAB   DC    C'0123456789ABCDEF'
HEXNC    DC    X'0F0F0F0F0F0F0F0F'
*
MSGRET   DS    0F
          DC    Y(EMSGRET-MSGRET)
          DS    CL3
          DC    C'PRNTDOC ERROR RETURN CODE : ,
APIRET   DS    CL8
          DC    C' FOR ,
TESTNR   DS    CL30
EMSGRET  EQU    *
*
          EJECT
          PRINT GEN
PLL      PRNTDOC MF=L
          PRINT NOGEN
*
          END
```

## 2.1 Die C-Schnittstelle für die Funktionen des Makros PRNTDOC

Die C-Schnittstelle realisiert die gleiche Funktionalität wie der Assembler-Makro PRNTDOC. Die einzelnen Funktionen können in analoger Weise mit C-Sprachmitteln spezifiziert werden. Die Kommentierung des Headerfiles "PRNTDOC.H" beschränkt sich deshalb häufig auf Verweise auf die Operandenbeschreibung des Assembler-Makros PRNTDOC.

Der Headerfile sowie der Funktionsaufruf (SVC-Aufruf) befinden sich in der Include-Datei "PRNTDOC.H". Diese Datei steht in der SPOOL-Standardbibliothek, die zum Lieferumfang von SPOOL ab V3.0 gehört.

Die Include-Datei gliedert sich in die folgenden Teile:

- Returncodes (in dezimaler Form)
- Konstantendefinitionen für bestimmte Funktionsbereiche
- Komplexere Datenstrukturen ("Records")
- Parameterliste mit Verweisen auf die komplexeren Datenstrukturen
- Funktionsaufruf

Die Datei "PRNTDOC.H" wird in einem C-Anwenderprogramm über einen Include-Aufruf mit dem Anwenderprogramm gebunden und die Parameterliste bereitgestellt.

### Schnittstellendefinition *prntdoc.h*

```
#ifndef _PRNTDOC_H
#define _PRNTDOC_H

#if 0
/*****
BEGIN-INTERFACE    PRNTDOC

TITLE              (/ prntdoc /)
NAME               PRNTDOC.H
DOMAIN            SPOOL
LANGUAGE          C
COPYRIGHT         (C) Siemens AG 1998
                  ALL RIGHTS RESERVED

COMPILATION-SCOPE USER
INTERFACE-TYPE    CALL
RUN-CONTEXT      TU,
                  TPR

PURPOSE           (/ Print-job management /)

-----
```



```

END-INTERFACE      PRNTDOC.
*****
#endif

/* main return codes */
/* mret_code */
#define PRNTDOCok 0 /* successfull processing */
#define PRNTDOCdelete_ignored 72 /* warning */
#define PRNTDOCinvalid_filename 1 /* invalid filename */
#define PRNTDOCinvalid_access 2 /* invalid pl access */
#define PRNTDOCinconsistency 8 /* inconsistency between */
/* param */
#define PRNTDOCinvalid_param 36 /* invalid parameter spec */
#define PRNTDOCwrong_recform 48 /* recform=F not allowed with */
/* recsize=0 */
#define PRNTDOCinteract_job 120 /* print *sysout not allowed */
#define PRNTDOCwrong_spacing 124 /* l-spac=ebcdic and recform */
/* U */
#define PRNTDOCfgg_not_allowed 128 /* fgg not allowed */
#define PRNTDOCwrong_HP_param 144 /* invalid or missing */
/* parameters for HP-HP90 */
#define PRNTDOCsf_dummy 152 /* system file assigned to */
/* *dummy */
#define PRNTDOCsf_tape 156 /* system file assigned to */
/* tape */
#define PRNTDOCsf_temporary 160 /* system file assigned to */
/* temporary one */
#define PRNTDOCsf_not_sam 164 /* system file not assigned */
/* to SAM file */
#define PRNTDOCnot_plam_lib 220 /* not a plam library */
#define PRNTDOCwrong_version 65535 /* interface version not */
/* supported */
#define PRNTDOCreqm_error 28 /* memory request error */
#define PRNTDOCrelnm_error 32 /* memory release error */
#define PRNTDOCqueue_error 4 /* queue routine error */
#define PRNTDOCno_info 212 /* no information on user job */
#define PRNTDOCinternal_error 252 /* internal error */
#define PRNTDOCdms_error 12 /* DMS error */
#define PRNTDOCeam_proc_error 16 /* error in EAM processing */
#define PRNTDOCwrong_password 24 /* invalid or missing */
/* password */
#define PRNTDOCinvalid_file_access 44 /* invalid file access */
#define PRNTDOCdate_not_available 52 /* creation-expiration date */
/* not available */
#define PRNTDOCeam_not_available 64 /* EAM file not available */
#define PRNTDOCno_data 68 /* syst file contains no data */
#define PRNTDOCdelete_active 84 /* DELETE-FILE command active */
#define PRNTDOCsf_close_error 108 /* syst file close error */

```

```

#define PRNTDOCfile_in_use 112      /* file currently in use      */
#define PRNTDOCdest_not_found 136  /* destination does not exist */
#define PRNTDOCf_access_not_avail 168 /* file access not available */
#define PRNTDOCform_not_found 180  /* form does not exist       */
#define PRNTDOCsam_error 184       /* SAM error in EAM          */
                                   /* processing                 */
#define PRNTDOCno_rotation 188     /* no rotation supported     */
#define PRNTDOCwrong_res_value 192 /* exceeding char-set-num or */
                                   /* FOB size                   */
#define PRNTDOCwrong_cs_index 196  /* exceeding char-set-num in */
                                   /* pool                       */
#define PRNTDOCno_rotation_loop 200 /* no rotation-loop defined  */
#define PRNTDOCno_tsn_assigned 204 /* new TSN can not be       */
                                   /* assigned                   */
#define PRNTDOCplam_lib_not_avail 224 /* PLAM library not available */
#define PRNTDOCplam_elem_not_found 228 /* library element not found */
#define PRNTDOCmonjv_error 232     /* monjv error              */
#define PRNTDOCauthorization_error 236 /* authorization error      */
#define PRNTDOCno_dist_allowed 240 /* distribution not allowed  */
#define PRNTDOCspool_pended 148    /* SPOOL subsystem pended   */
#define PRNTDOCsubsys_not_avail 65535 /* S&P subsystems not     */
                                   /* available                  */

/* _dtype_set */
/* ENUM _dtype_set */
#define PRNTDOCdtype_file 1         /* print a file              */
#define PRNTDOCdtype_libel 2       /* print a library-element   */
#define PRNTDOCdtype_posix 3       /* print a POSIX-file       */
#define PRNTDOCdtype_eam 4         /* print a EAM-file         */
#define PRNTDOCdtype_sysf 5       /* print a system file      */

/* _family_set */
/* ENUM _family_set */
#define PRNTDOCfamily_std 1         /* family = *std            */
#define PRNTDOCfamily_yes 2        /* family = *yes           */
#define PRNTDOCfamily_no 3         /* family = *no            */

/* _checkp_set */
/* ENUM _checkp_set */
#define PRNTDOCon_pages 0          /* checkp = *on-page       */
#define PRNTDOCon_section_records 1 /* checkp =                */
                                   /* *on-section-record     */

/* _lockf_set */
/* ENUM _lockf_set */
#define PRNTDOClockf_std 0          /* lockf = *std            */
#define PRNTDOClockf_no 1          /* lockf = *no             */
#define PRNTDOClockf_yes 2         /* lockf = *yes            */

```

```

/* _delf_set */
/* ENUM _delf_set */
#define PRNTDOCdelf_yes 1 /* delf = *yes */
#define PRNTDOCdelf_destr 2 /* delf = *destroy */
#define PRNTDOCdelf_no 3 /* delf = *no */

/* _outform_set */
/* ENUM _outform_set */
#define PRNTDOCout_character 1 /* outform = *character */
#define PRNTDOCout_hexadecimal 2 /* outform = *hexadecimal */

/* _twoside_set */
/* ENUM _twoside_set */
#define PRNTDOctwoside_std 0 /* twoside = *std */
#define PRNTDOctwoside_yes 1 /* twoside = *yes */
#define PRNTDOctwoside_tumble 2 /* twoside = *tumble */
#define PRNTDOctwoside_no 4 /* twoside = *no */
#define PRNTDOctwoside_ignore 8 /* twoside = *ignore */

/* _rotation_set */
/* ENUM _rotation_set */
#define PRNTDOCrotation_no 213 /* rot = *no */
#define PRNTDOCrotation_0 240 /* rot = *rot_0 */
#define PRNTDOCrotation_90 241 /* rot = *rot_90 */
#define PRNTDOCrotation_180 242 /* rot = *rot_180 */
#define PRNTDOCrotation_270 243 /* rot = *rot_270 */
#define PRNTDOCrotation_0_180 244 /* rot = *rot_0_180 */
#define PRNTDOCrotation_90_270 245 /* rot = *rot_90_270 */
#define PRNTDOCrotation_180_0 246 /* rot = *rot_180_0 */
#define PRNTDOCrotation_270_90 247 /* rot = *rot_270_90 */
#define PRNTDOCby_control_codes 232 /* rot = *by_control_codes */

/* _intray_set */
/* ENUM _intray_set */
#define PRNTDOCintray_by_format 0 /* intray = *by-format */
#define PRNTDOCintray_ignore 255 /* intray = *ignore */

/* _intray_format_set */
/* ENUM _intray_format_set */
#define PRNTDOCitf_manual 128 /* intray_format = *manual */
#define PRNTDOCitf_a3 129 /* intray_format = *a3 */
#define PRNTDOCitf_a4 130 /* intray_format = *a4 */
#define PRNTDOCitf_a5 131 /* intray_format = *a5 */
#define PRNTDOCitf_b4 132 /* intray_format = *b4 */
#define PRNTDOCitf_b5 133 /* intray_format = *b5 */
#define PRNTDOCitf_double_letter 134 /* intray_format =
/* *double-letter */

```

```

#define PRNTDOCitf_exec 135          /* intray_format = *exec */
#define PRNTDOCitf_folio 136        /* intray_format = *folio */
#define PRNTDOCitf_invoice 137      /* intray_format = *invoice */
#define PRNTDOCitf_legal 138        /* intray_format = *legal */
#define PRNTDOCitf_letter 139       /* intray_format = *letter */
#define PRNTDOCitf_monarch 140      /* intray_format = *monarch */
#define PRNTDOCitf_commercial_10 141 /* intray_format = */
                                   /* *commercial-10 */
#define PRNTDOCitf_d1 142           /* intray_format = *d1 */
#define PRNTDOCitf_c5 143           /* intray_format = *c5 */
#define PRNTDOCitf_A3U 144          /* intray_format = *A3-uncut */
#define PRNTDOCitf_A4U 145          /* intray_format = *A4-uncut */
#define PRNTDOCitf_ledger 146       /* intray_format = *ledger */

/* _outtray_set */
/* ENUM _outtray_set */
#define PRNTDOCouttray_ignore 128   /* outtray = *ignore */
#define PRNTDOCouttray_sorter 129   /* outtray = *sorter */

/* _srtmode_set */
/* ENUM _srtmode_set */
#define PRNTDOCsrtmode_no 1         /* srtmode = *no */
#define PRNTDOCsrtmode_group 2      /* srtmode = *group */
#define PRNTDOCsrtmode_collate 3    /* srtmode = *collate */
#define PRNTDOCsrtmode_stacker 4    /* srtmode = *stacker */
#define PRNTDOCsrtmode_automatic 5  /* srtmode = *automatic */

/* _docform_set */
/* ENUM _docform_set */
#define PRNTDOCdocform_text 0        /* docform = *text */
#define PRNTDOCdocform_page_format 1 /* docform = *page-format */
#define PRNTDOCdocform_spec_format 2 /* docform = *special-format */

/* _control_mode_set */
/* ENUM _control_mode_set */
#define PRNTDOCctl_mode_std 1        /* mode = *std */
#define PRNTDOCctl_mode_phys 2       /* mode = *physical */
#define PRNTDOCctl_mode_page_m 4     /* mode = *page-mode */
#define PRNTDOCctl_mode_logic 8      /* mode = *logical */
#define PRNTDOCctl_mode_line_m 16    /* mode = *line-mode */
#define PRNTDOCctl_mode_apa 64       /* mode = *apa */

/* _pcc_set */
/* ENUM _pcc_set */
#define PRNTDOCtop_mode_yes 0        /* pcc = *yes */
#define PRNTDOCtop_mode_no 1         /* pcc = *no */
#define PRNTDOCtop_mode_dummy 2      /* pcc = *dummy */

```

```

/* _stproc_set */
/* ENUM _stproc_set */
#define PRNTDOCstproc_value 0 /* stproc = value TPR only */
#define PRNTDOCstproc_no 1 /* stproc = no TPR only */
#define PRNTDOCstproc_close 2 /* stproc = close TPR only */
#define PRNTDOCstproc_imm 4 /* stproc = immediate TPR only */

/* _space_set */
/* ENUM _space_set */
#define PRNTDOCspace_1 1 /* spacing = *space_1 */
#define PRNTDOCspace_2 2 /* spacing = *space_2 */
#define PRNTDOCspace_3 4 /* spacing = *space_3 */
#define PRNTDOCspace_e 8 /* spacing =
/* *by_ebcdic_control */
#define PRNTDOCspace_a 16 /* spacing = *by_asa_control */
#define PRNTDOCspace_i 32 /* spacing = *by_ibm_control */
#define PRNTDOCspace_no 144 /* spacing = *no */
#define PRNTDOCspace_std 0 /* spacing = *std */

/* _control_type_set */
/* ENUM _control_type_set */
#define PRNTDOCctl_type_dummy 0 /* ctltype = *dummy */
#define PRNTDOCctl_type_hp_com 1 /* ctltype = *compatible */
#define PRNTDOCctl_type_hp 2 /* ctltype = *hp */

/* _headline_set */
/* ENUM _headline_set */
#define PRNTDOHeadline_no 1 /* headlin = *no */
#define PRNTDOHeadline_std 2 /* headlin = *std */
#define PRNTDOHeadline_r 4 /* headlin = *frecord */
#define PRNTDOHeadline_d 8 /* headlin = *date */
#define PRNTDOHeadline_dr 12 /* headlin = *date_frecord */
#define PRNTDOHeadline_p 16 /* headlin = *page */
#define PRNTDOHeadline_rp 20 /* headlin = *frecord_page */
#define PRNTDOHeadline_dp 24 /* headlin = *date_page */
#define PRNTDOHeadline_drp 28 /* headlin =
/* *date_frecord_page */

/* _truncation_set */
/* ENUM _truncation_set */
#define PRNTDOctrunc_std 0 /* trunc = *std */
#define PRNTDOctrunc_delete_file 1 /* trunc = *delete_file */
#define PRNTDOctrunc_keep_file 2 /* trunc = *keep_file */

```

```

/* _layout_part_dim_set */
/* ENUM _layout_part_dim_set */
#define PRNTDOClayout_none 0 /* dim = *none */
#define PRNTDOClayout_pages 1 /* dim = *pages */
#define PRNTDOClayout_lines 2 /* dim = *line */

/* _prtype_set */
/* ENUM _prtype_set */
#define PRNTDOCprtype_any 0 /* prtype = *any */
#define PRNTDOCprtype_lp65 1 /* prtype = *lp65_printer */
#define PRNTDOCprtype_apa 2 /* prtype = *apa_printer */
#define PRNTDOCprtype_hp 3 /* prtype = *hp_printer */

/* _virtual_set */
/* ENUM _virtual_set */
#define PRNTDOCvirtual_must 4 /* virtual = *must */
#define PRNTDOCvirtual_allo 2 /* virtual = *allowed */
#define PRNTDOCvirtual_nota 1 /* virtual = *not_allowed */
#define PRNTDOCvirtual_stda 0 /* virtual = *std */

/* _libelem_md1 */
struct PRNTDOC_libelem_md1 {
    char element[80]; /* element name */
    char version[40]; /* element version */
    char type[12]; /* element type */
};

/* _sectrec_md1 */
struct PRNTDOC_sectrec_md1 {
    char sect_id[60]; /* section id */
    unsigned short sect_pos; /* section id position */
    unsigned char sect_len; /* section id length */

    /* sect_ind */
    struct /* sect_ind */ {
        char sect_given: 1; /* This bit must be set if */
        /* sect-id is given, i.e. if */
        /* it is not *NONE. */
        char input_char: 1; /* section-id is given in */
        /* character (= true) or */
        /* hexadecimal (= false). */
        /* This field serves to */
        /* indicate in which format */
        /* the section-id should be */
        /* printed on the */
        /* trailer-page. */
        char sectrec_free: 6; /* not used */
    } sect_ind;
};

```

```

};

/* _firsrec_md1 */
struct PRNTDOC_firsrec_md1 {
    unsigned long stpos;          /* record number */
    char ststrid[60];           /* string-id */
    unsigned short ststrpos;     /* string-id position */
    unsigned short ststrocc;     /* string-id occurrence */
    unsigned char ststrlen;      /* string-id length */
    char filler6[1];            /* not used */

    /* strid_ind */
    struct /* strid_ind */ {
        unsigned short st_char: 1;
        /* string-id is given in
        /* character (= true) or
        /* hexadecimal (= false).
        /* This field serves to
        /* indicate in which format
        /* the section-id should be
        /* printed on the
        /* trailer-page.
        unsigned short ststrid_free: 15;
        /* not used
    } strid_ind;
};

/* _lastrec_md1 */
struct PRNTDOC_lastrec_md1 {
    unsigned long enpos;          /* last record number */
    char enstrid[60];           /* string-id */
    unsigned short enstrpos;     /* string-id position */
    unsigned short enstrocc;     /* string-id occurrence */
    unsigned char enstrlen;      /* string-id length */
    char filler7[1];            /* not used */

    /* strid_ind */
    struct /* strid_ind */ {
        unsigned short en_char: 1;
        /* string-id is given in
        /* character (= true) or
        /* hexadecimal (= false).
        /* This field serves to
        /* indicate in which format
        /* the section-id should be
        /* printed on the
        /* trailer-page.
        unsigned short enstrid_free: 15;

```

```

/* not used */
    } strid_ind;
};

/* _recpart_md1 */
struct PRNTDOC_recpart_md1 {
    unsigned short first_ch; /* first character */
    unsigned short last_ch; /* last character */
};

/* _linesp_md1 */
struct PRNTDOC_linesp_md1 {
    unsigned short cc_pos; /* control char position */
    unsigned char spacing; /* line spacing */
    char filler4[1]; /* not used */
};

/* _contmod_md1 */
struct PRNTDOC_contmod_md1 {
    unsigned char ctlmode; /* control mode */
    unsigned char pcc; /* page control character */
    unsigned char ctltype; /* control type */
};

/* _procadm_md1 */
struct PRNTDOC_procadm_md1 {
    char user[8]; /* user-id under which the */
    /* print-job is launched */
    char account[8]; /* account number under which */
    /* the print-job is launched */
    char password[8]; /* password for the user-id */
    /* given with the <user> */
    /* parameter */
};

/* _outpart_md1 */
struct PRNTDOC_outpart_md1 {
    long rec_from; /* from record */
    unsigned long rec_to; /* till record */
    unsigned char dim; /* dimension */
    char filler5[3]; /* not used */
};

/* _covpage_md1 */
struct PRNTDOC_covpage_md1 {
    char headtext[32]; /* header page text */
    long headexit; /* header exit number */
    long traiaexit; /* trailer exit number */
};

```



```

};

/* _overlay_md1 */
struct PRNTDOC_overlay_md1 {
    unsigned char face;          /* face side */
    unsigned char reverse;      /* reverse side */
};

/* _transtab_md1 */
struct PRNTDOC_transtab_md1 {
    char name[8];               /* translation table name */
    char file[44];             /* translation table file */
};

/* _transtab_md1 */
struct PRNTDOC_virtual_md1 {
    char name[8];              /* virtual program name */
    char string[32];          /* virtual program string */
};

/* parameter list description */
struct PRNTDOC_pl_md1 {
    /* Standard header */
    struct ESMFHDR hdr;
    void* rsopar;              /* address of rso_pl */
    void* spspar;              /* address of sps_pl */
    void* dppar;               /* address of dprint_pl */
    void* dsempar;            /* address of dsem_pl */
    void* reserved_2;          /* free space */
    void* reserved_3;          /* free space */
    void* reserved_4;          /* free space */
    unsigned short dtype;      /* type of the document to
                               /* print
    unsigned short dname_len;  /* Length of the name of the
                               /* document(s) to be printed
    void* dname_ptr;           /* Address of the name of the
                               /* document(s) to be printed
    unsigned long eamn[16];    /* EAM-file-number(s) to be
                               /* printed
    unsigned short sysnn[16]; /* *SYSOUT/SYSLSTnn to be
                               /* printed
    struct PRNTDOC_libelem_md1 libelem;
                               /* Library element(s) to be
                               /* printed
    struct PRNTDOC_sectrec_md1 sectrec;
                               /* Idenfier of the optional
                               /* section records

```

```

struct PRNTDOC_firsrec_md1 firsrec;
    /* Beginning part of the      */
    /* document(s) to be printed */
struct PRNTDOC_lastrec_md1 lastrec;
    /* End part of the            */
    /* document(s) to be printed */
struct PRNTDOC_recpart_md1 recpart;
    /* Part of the input record   */
    /* to be processed           */
struct PRNTDOC_linesp_md1 linesp;
    /* Number of line feeds after */
    /* a line has been printed    */
unsigned char docform;          /* document format      */
struct PRNTDOC_contmod_md1 contmod;
    /* Control mode applied by   */
    /* the Spool & Print        */
    /* subsystem on the         */
    /* document(s) copies      */
unsigned char family;          /* a common TSN is allocated */
    /* if a number of files or  */
    /* library elements are     */
    /* specified                */
unsigned char checkp;          /* Checkpoint processing is */
    /* to be performed by the   */
    /* controller on the basis of */
    /* pages or section         */
unsigned short doc_number;     /* Number of files        */
long stproc_val;              /* ONLY TPR: Time of printout */
    /* for the system file prior */
    /* to termination of the job */
unsigned char stproc;          /* ONLY TPR: specifies    */
    /* whether time of printout  */
    /* for the system file is   */
    /* prior to termination of  */
    /* the job                  */
unsigned char lockf;           /* Lock while job is waiting */
unsigned char delf;           /* Delete (destroy) document */
unsigned char addcop;          /* how many additional times */
    /* the print-job must be    */
    /* repeated (number of     */
    /* additional document(s)  */
    /* copies)                  */

/* jvpass                                                                */
union /* jvpass */ {
    long jvpassi;                /* Password with which the */
    /* job variable is protected */
    /* (numeric)                  */

```

```

        char jvpassw[4];          /* Password with which the      */
                                  /* job variable is protected    */
    } jvpass;
    char monjv[54];              /* Job variable into which      */
                                  /* information on print-job     */
                                  /* processing is to be stored   */
    char prjname[8];            /* Job-name of the print-job    */
    unsigned char prjprio;      /* Priority of the print-job     */
    struct PRNTDOC_procadm_md1 procadm;
                                  /* Allows to execute the       */
                                  /* print-job under another     */
                                  /* user-id                      */
    unsigned char truncation;   /* Line truncation              */
    struct PRNTDOC_outpart_md1 outpart;
                                  /* layout control (number of   */
                                  /* pages or lines)             */
    unsigned char prjclas;      /* job class                    */
    unsigned char leftmarg;     /* Indentation for the output   */
                                  /* text                         */
    unsigned short linepp;      /* Lines per page (including    */
                                  /* header and blanks)          */
    unsigned char headline;     /* Type of header line         */
    unsigned char outform;      /* The output format is        */
                                  /* character format only or     */
                                  /* character and hexadecimal    */
                                  /* format                       */
    unsigned char twoside;      /* The document is to be      */
                                  /* printed on one or both      */
                                  /* paper sides (supported      */
                                  /* only on LP65 printers)     */
    unsigned char rotation;     /* The pages to be printed     */
                                  /* from the print-job are     */
                                  /* rotated                      */

    /* intray                                                              */
    union /* intray_union */ {
        unsigned char intray; /* Paper input tray to be     */
                                  /* used (only on LP65 and RS0  */
                                  /* printers)                   */
        unsigned char in_tray; /*
        unsigned char intray_format; /*
                                  /*
    } intray_union;

    /* outtray                                                              */
    union /* outtray_union */ {
        unsigned char outtray; /* Paper output tray to be   */
                                  /* used (only on LP65 and RS0 */

```

```

/* printers) */
    unsigned char out_tray; /* */
} outtray_union;
short pagecop; /* Copies of each printed */
/* page */
struct PRNTDOC_covpage_md1 covpage;
/* Cover-page */
char form[6]; /* Form-name to be used */
char loop[3]; /* Loop-name to be used */
char rotloop[3]; /* Rotation-loop name to be used */
/* used */
char charset[16][3]; /* One or more character-sets */
/* used for printing */
char poolnam[4]; /* Pool-name */
unsigned char poolind; /* Pool-index */
char efo[2]; /* Film or EFO overlay for */
/* printing */
struct PRNTDOC_overlay_md1 overlay;
/* Double sided overlay (only */
/* for LP65 printer type) */
char fob[4]; /* FOB (Forms Overlay Buffer) */
/* data overlay to be used */
/* for printing the document */
/* (only supported on HP */
/* printer equipped with a */
/* graphic buffer */
unsigned char virtual_s; /* allow virtual controller */
char filler2[2]; /* slack bytes */
unsigned long pagepcl; /* PCL file number (only for */
/* LP65 printer type) */
char userres[44]; /* User resources file to be */
/* used for print resources */
/* (user PRFILE generated by */
/* PRM) */
struct PRNTDOC_transtab_md1 transtab;
/* Translation table to be */
/* used */
char pname[8]; /* Printer or printer pool */
/* name on which the document */
/* is to be printed */
unsigned char prtype; /* Printer type requested */
unsigned char srtmode; /* Sorter mode */
char filler3[2]; /* slack bytes */
char tsn[4]; /* output: TSN obtained from */
/* the system */
char cluster[8]; /* for compatibility with */
/* MODIFY-PRINT-JOB */

```

```
/* specified1 */
struct /* specified1 */ {
    char spec1_dtype: 1; /* document type */
    char spec1_dname: 1; /* document name */
    char spec1_dnum: 1; /* document number */
    char spec1_libelem: 1; /* library element */
    char spec1_sectrec: 1; /* section record */
    char spec1_firsrec: 1; /* first record */
    char spec1_lastrec: 1; /* last record */
    char spec1_recpart: 1; /* record part */
} specified1;

/* specified2 */
struct /* specified2 */ {
    char spec2_linesp: 1; /* line spacing */
    char spec2_docform: 1; /* document format */
    char spec2_contmod: 1; /* control mode */
    char spec2_addcop: 1; /* additional copies */
    char spec2_family: 1; /* family */
    char spec2_checkp: 1; /* checkpoint */
    char spec2_monjv: 1; /* monjv */
    char spec2_jvpassw: 1; /* jv-password */
} specified2;

/* specified3 */
struct /* specified3 */ {
    char spec3_prjname: 1; /* print-job-name */
    char spec3_prjprio: 1; /* print-job-priority */
    char spec3_procadm: 1; /* processing admission */
    char spec3_trunc: 1; /* line-truncation */
    char spec3_lockf: 1; /* lock-file */
    char spec3_delf: 1; /* delete-after-print */
    char spec3_outpart: 1; /* output part */
    char spec3_pagecop: 1; /* page-copies */
} specified3;

/* specified4 */
struct /* specified4 */ {
    char spec4_leftmar: 1; /* left-margin */
    char spec4_linepp: 1; /* line-per-page */
    char spec4_headlin: 1; /* header-line */
    char spec4_outform: 1; /* output-format */
    char spec4_twoside: 1; /* two-sided */
    char spec4_rot: 1; /* rotation */
    char spec4_intray: 1; /* input-tray */
    char spec4_outtray: 1; /* output-tray */
} specified4;
```

```

/* specified5 */
struct /* specified5 */ {
    char spec5_covpage: 1; /* cover-page */
    char spec5_form: 1; /* form-name */
    char spec5_loop: 1; /* loop-name */
    char spec5_rotloop: 1; /* rotation-loop-name */
    char spec5_charset: 1; /* character-set */
    char spec5_poolnam: 1; /* pool-name */
    char spec5_poolind: 1; /* pool-index */
    char spec5_efo: 1; /* electronic-form-overlay */
} specified5;

/* specified6 */
struct /* specified6 */ {
    char spec6_overlay: 1; /* overlay */
    char spec6_fob: 1; /* fob */
    char spec6_pagepcl: 1; /* lp65-pagedef */
    char spec6_userres: 1; /* user resources file */
    char spec6_transtab: 1; /* translation table */
    char spec6_prname: 1; /* printer-name */
    char spec6_prtype: 1; /* printer-type */
    char spec6_tsn: 1; /* tsn */
} specified6;

/* specified7 */
struct /* specified7 */ {
    char spec7_rsopar: 1; /* rso parameters */
    char spec7_spspar: 1; /* sps parameters */
    char spec7_dppar: 1; /* dprint parameters */
    char spec7_dsempar: 1; /* dsem parameters */
    char spec7_free: 3; /* */
} specified7;

/* specified8 */
struct /* specified8 */ {
    char spec8_stproc: 1; /* start-processing */
    char spec8_srtmode: 1; /* sorter mode */
    char spec8_virtual: 1; /* virtual control */
    char spec8_jclass: 1; /* job-class */
    char spec8_prpname: 1; /* prog. name */
    char spec8_strprog: 1; /* string field */
    char spec8_frmname: 1; /* format name */
    char spec8_free: 1; /* */
} specified8;

```

```

struct PRNTDOC_virtual_md1 program;
                                /* virtual device synchrone */
char format_name[63];           /* format name */
char mod_res[2];                /* reserved for modify */

/* specified9 */
struct /* specified9 */ {
    char spec9_topoff: 1;       /* top-offset */
    char spec9_leftoff: 1;     /* left-offset */
    char spec9_schedtime: 1;   /* scheduling time */
    char spec9_free: 5;        /*
} specified9;
short top_offset;              /* Specifies the length of
                                /* the white band that must
                                /* be inserted
short left_offset;            /* Specifies the width of the
                                /* white band that must be
                                /* inserted
char schdate[10];             /* scheduling date
char schtime[5];              /* scheduling time
char unused9[11];             /* unused
};

/* call PRNTDOC via svc 108 */
#ifdef __SNI_HOST_BS2000
void _SVC(int, void*);
#define PRNTDOC(p) _SVC(108, &p)
#endif

#endif /* _PRNTDOC_H */

```

## 2.2 Die COBOL-Schnittstelle für die Funktionen des Makros PRNTDOC

Die COBOL-Schnittstelle realisiert die gleiche Funktionalität wie der Assembler-Makro PRNTDOC. Die einzelnen Funktionen können in analoger Weise mit COBOL-Sprachmitteln spezifiziert werden. Die Kommentierung der nachfolgend aufgeführten Datenstrukturen beschränkt sich deshalb häufig auf Verweise auf die Operandenbeschreibung des Assembler-Makros PRNTDOC.

Die Datenstrukturen sowie der Funktionsaufruf befinden sich im Runtime-Modul SCPADDR. Dieser Modul gehört zum Lieferumfang von SPOOL ab Version 3.0.

Die Schnittstelle wird in COBOL über eine Call-Anweisung mit Übergabe eines Parameterbereiches aufgerufen. Die nachfolgend beschriebenen Felder dieser Schnittstelle können mit der Copy-Anweisung innerhalb des Working-Storage-Struktur in den COBOL-Quelltext kopiert werden (siehe auch das Beispiel ab [Seite 172](#)).

Das COBOL-Anwenderprogramm muss mit dem Runtime-Modul SCPADDR gelinkt werden. In diesem Modul sind die folgenden Funktionen für den Anwender verfügbar:

- Funktion SCPADDR : modifiziert den Datenbereich "param-1" mit der in "param-2" übergebenen Adresse.
- Funktion SCPSETB : setzt die Bits des Datenbereichs " param-2" im Datenbereich "param-1" mit der in "param-3" angegebenen Länge.
- Funktion SCPSETB1 : setzt die Bits des Datenbereichs " param-2" im Datenbereich "param-1" mit der Länge 1 Byte.
- Funktion SCPRSTB : setzt die Bits des Datenbereichs " param-2" im Datenbereich "param-1" mit der in "param-3" angegebenen Länge zurück.
- Funktion SCPRSTB1 : setzt die Bits des Datenbereichs " param-2" im Datenbereich "param-1" mit Länge 1 Byte zurück.
- Funktion PRNTDOC: stößt einen Druckauftrag mit den in den Datenbereichen angegebenen Parametern an.

Für jedes Feld, in dem kein Wert, sondern eine Adresse hinterlegt werden soll, muss die Funktion SCPADDR aufgerufen werden, um die benötigte Adresse einzutragen. Dieser Aufruf ist verpflichtend für das Dateinamens-Feld und für das Feld mit der Adresse der Parameterliste für die Subsysteme RSO, DPRINT, SPS oder DSEM.

Für jeden benötigten Operanden muss durch Aufruf der Funktionen SCPSETB oder SCPSETB1 ein Flag gesetzt werden. Wenn jeder Operand mit dem entsprechenden Wert versorgt und das Flag gesetzt ist, kann die Funktion PRNTDOC (über Call-Anweisung) aufgerufen werden, um einen Druckauftrag anzustoßen.



**Beschreibung der Parameterliste**

\*\*\*\*\*

\* PRNTDOCY - Declarations

\*\*\*\*\*

\* yes-no-set

01 PRNTDOC-yes-no-set.

02 PRNTDOC-yes-no-yes PIC X(01) VALUE SYS-X01.

02 PRNTDOC-yes-no-no PIC X(01) VALUE SYS-X02.

\* doc-part-type-set

**01 PRNTDOC-doc-part-type-set.**

<< Datenstruktur, die alle Angaben für einen Anfangs und Enddatensatz >>

<< zusammenfasst, wenn von der Eingabedatei nur eine bestimmte Anzahl >>

<< von Datensätzen verarbeitet werden soll. >>

<< Siehe auch Beschreibung der Operanden "FIRSREC" auf [Seite 64](#) und >>

<< "LASTREC" auf [Seite 76](#). >>

02 PRNTDOC-doc-part-begin-of-file PIC S9(09) COMP  
VALUE 1.

02 PRNTDOC-doc-part-eof-char.

03 FILLER PIC X(1)  
VALUE SYS-X7F.

03 FILLER PIC X(3)  
VALUE HIGH-VALUES.

02 PRNTDOC-doc-part-end-of-file REDEFINES  
PRNTDOC-doc-part-eof-char PIC S9(09) COMP.

02 PRNTDOC-doc-part-by-string-id PIC S9(09) COMP  
VALUE ZERO.

\* sect-ind

**01 PRNTDOC-sect-ind-bits.**

<< Datenstruktur, in der die Angaben zusammengefasst sind, ob die >>

<< Eingabedatei in Abschnittsmarken aufgeteilt ist und ob die Marken >>

<< im Zeichenformat oder hexadezimal vorliegen. >>

<< Siehe auch Beschreibung des Operanden "SECTREC" auf [Seite 102](#). >>

02 PRNTDOC-sect-id-given PIC X(01).

88 PRNTDOC-sect-id-given-yes VALUE SYS-X80.

88 PRNTDOC-sect-id-given-no VALUE SYS-X00.

02 PRNTDOC-sect-id-input PIC X(01).

88 PRNTDOC-sect-id-input-char VALUE SYS-X40.

88 PRNTDOC-sect-id-input-hex VALUE SYS-X00.

\* prtype-set

**01 PRNTDOC-prtype-set PIC X(01).**

<< Datenstruktur, in der über Bedingungsnamen-Definition die für den >>

<< Ausdruck möglichen Druckertypen zusammengefasst sind, >>

<< Siehe auch die Beschreibung des Operanden "PRTYPE" auf [Seite 96](#). >>

88 PRNTDOC-lp65-printer VALUE SYS-X04.

88 PRNTDOC-hp-printer VALUE SYS-X02.

88 PRNTDOC-apa-printer VALUE SYS-X01.

**01 PRNTDOC-specified-set.**

```
<< Datenstruktur, in der die Menge der möglichen Optionen (= Operanden >>
<< des Assembler-Makros PRNTDOC), jeweils zu Blöcken (specified1 bis >>
<< specified7) zusammengefasst, aufgelistet ist. >>
<< Wird ein Operand aus einem der Blöcke für den SPOOLOUT-Auftrag benö- >>
<< tigt, muss der Zugriff auf dieses Feld durch Setzen des Bit-Indikators >>
<< (zugeordneter Feldname in der Parameterliste: PRNTDOC-specified; >>
<< Seite 166) ermöglicht werden. Die Feldnamen entsprechen i.w. den >>
<< Operandennamen des Assembler-Makros PRNTDOC. >>
```

```
02 PRNTDOC-specified1-set PIC X(01).
88 PRNTDOC-specified1-dtype VALUE SYS-X80.
88 PRNTDOC-specified1-dname VALUE SYS-X40.
88 PRNTDOC-specified1-dnum VALUE SYS-X20.
88 PRNTDOC-specified1-libelem VALUE SYS-X10.
88 PRNTDOC-specified1-sectrec VALUE SYS-X08.
88 PRNTDOC-specified1-firsrec VALUE SYS-X04.
88 PRNTDOC-specified1-lastrec VALUE SYS-X02.
88 PRNTDOC-specified1-recpart VALUE SYS-X01.
```

```
02 PRNTDOC-specified2-set PIC X(01).
88 PRNTDOC-specified2-linesp VALUE SYS-X80.
88 PRNTDOC-specified2-docform VALUE SYS-X40.
88 PRNTDOC-specified2-contmod VALUE SYS-X20.
88 PRNTDOC-specified2-addcop VALUE SYS-X10.
88 PRNTDOC-specified2-family VALUE SYS-X08.
88 PRNTDOC-specified2-checkp VALUE SYS-X04.
88 PRNTDOC-specified2-monjv VALUE SYS-X02.
88 PRNTDOC-specified2-jvpassw VALUE SYS-X01.
```

```
02 PRNTDOC-specified3-set PIC X(01).
88 PRNTDOC-specified3-prjname VALUE SYS-X80.
88 PRNTDOC-specified3-prjprio VALUE SYS-X40.
88 PRNTDOC-specified3-procadm VALUE SYS-X20.
88 PRNTDOC-specified3-trunc VALUE SYS-X10.
88 PRNTDOC-specified3-lockf VALUE SYS-X08.
88 PRNTDOC-specified3-delf VALUE SYS-X04.
88 PRNTDOC-specified3-outpart VALUE SYS-X02.
88 PRNTDOC-specified3-pagecop VALUE SYS-X01.
```

```
02 PRNTDOC-specified4-set PIC X(01).
88 PRNTDOC-specified4-leftmar VALUE SYS-X80.
88 PRNTDOC-specified4-linepp VALUE SYS-X40.
88 PRNTDOC-specified4-headlin VALUE SYS-X20.
88 PRNTDOC-specified4-outform VALUE SYS-X10.
88 PRNTDOC-specified4-twoside VALUE SYS-X08.
88 PRNTDOC-specified4-rot VALUE SYS-X04.
88 PRNTDOC-specified4-intray VALUE SYS-X02.
88 PRNTDOC-specified4-outtray VALUE SYS-X01.
```

```
02 PRNTDOC-specified5-set PIC X(01).
88 PRNTDOC-specified5-covpage VALUE SYS-X80.
88 PRNTDOC-specified5-form VALUE SYS-X40.
```

```
88 PRNTDOC-specified5-loop          VALUE SYS-X20.
88 PRNTDOC-specified5-rotloop       VALUE SYS-X10.
88 PRNTDOC-specified5-charset       VALUE SYS-X08.
88 PRNTDOC-specified5-poolnam       VALUE SYS-X04.
88 PRNTDOC-specified5-poolind       VALUE SYS-X02.
88 PRNTDOC-specified5-efo           VALUE SYS-X01.
02 PRNTDOC-specified6-set          PIC X(01).
88 PRNTDOC-specified6-overlay       VALUE SYS-X80.
88 PRNTDOC-specified6-fob           VALUE SYS-X40.
88 PRNTDOC-specified6-pagepcl       VALUE SYS-X20.
88 PRNTDOC-specified6-userres       VALUE SYS-X10.
88 PRNTDOC-specified6-trantab       VALUE SYS-X08.
88 PRNTDOC-specified6-prname        VALUE SYS-X04.
88 PRNTDOC-specified6-prtype        VALUE SYS-X02.
02 PRNTDOC-specified7-set          PIC X(01).
88 PRNTDOC-specified7-rsopar        VALUE SYS-X80.
88 PRNTDOC-specified7-spspar        VALUE SYS-X40.
88 PRNTDOC-specified7-dppar         VALUE SYS-X20.
88 PRNTDOC-specified7-dsempar       VALUE SYS-X10.
```

```
/-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----*
*   Parameter list description                                           *
*-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----*
```

**01 PRNTDOC-pl.**

```
<< Erzeugen der Parameterliste für den Funktionsaufruf      >>
```

```
*   Standard header
*   COPY fhdry.
```

```
*   main return codes
```

```
<< Definition der dezimalen Returncodes über Bedingungsnamen  >>
```

```
*   PRNTDOC-mret-code=SET.
```

```
88 PRNTDOC-mret-rc-ok                VALUE 0.
88 PRNTDOC-mret-warning               VALUE 1.
88 PRNTDOC-mret-recoverable-error     VALUE 2.
88 PRNTDOC-mret-fatal-error           VALUE 3.
88 PRNTDOC-mret-internal-error        VALUE 4.
```

**02 PRNTDOC-rso-pl** PIC S9(09) COMP.

```
<< Datenfeld zur Speicherung der Adresse einer Parameterliste, >>
```

```
<< in der RSO-spezifische Werte hinterlegt sind.           >>
```

```
*   address of rso-pl
```

```
88 PRNTDOC-rso-pl-none                VALUE -1.
```

**02 PRNTDOC-sps-pl** PIC S9(09) COMP.

```
<< Datenfeld zur Speicherung der Adresse einer Parameterliste, >>
```

```
<< in der SPS-spezifische Werte hinterlegt sind.           >>
```

```
*   address of sps-pl
```

```
88 PRNTDOC-sps-pl-none                VALUE -1.
```

```

02 PRNTDOC-dprint-pl PIC S9(09) COMP.
<< Datenfeld zur Speicherung der Adresse einer Parameterliste, >>
<< in der Dprint-spezifische Werte hinterlegt sind. >>
* address of dprint-pl
  88 PRNTDOC-dprint-pl-none VALUE -1.

02 PRNTDOC-dsem-pl PIC S9(09) COMP.
<< Datenfeld zur Speicherung der Adresse einer Parameterliste, >>
<< in der DSEM-spezifische Werte hinterlegt sind. >>
* address of dsem-pl
  88 PRNTDOC-dsem-pl-none VALUE -1.
  02 PRNTDOC-reserved-1 PIC S9(09) COMP.
  02 PRNTDOC-reserved-2 PIC S9(09) COMP.
  02 PRNTDOC-reserved-3 PIC S9(09) COMP.
  02 PRNTDOC-reserved-4 PIC S9(09) COMP.

02 PRNTDOC-dtype PIC 9(04) COMP.
<< In diesem strukturierten Datenfeld werden die möglichen Typen an >>
<< auszugebenden Dateien definiert. Jedem Bedingungsnamen ist ein ganz- >>
<< zahliger Wert zugeordnet. >>
<< Zur Beziehung zwischen Wert und Dateityp siehe auch die Beschreibung >>
<< des Operanden "DTYPE" auf Seite 61. >>
* dtype-set
  88 PRNTDOC-dtype-file VALUE 1.
* print a file
  88 PRNTDOC-dtype-lib-elem VALUE 2.
* print a library-element
  88 PRNTDOC-dtype-posix-path VALUE 3.
* print a POSIX-file
  88 PRNTDOC-dtype-eamn VALUE 4.
* print a EAM-file
  88 PRNTDOC-dtype-system-file VALUE 5.
* print a system-file

02 PRNTDOC-dname.
<< In diesem Datenfeld kann die Länge des Dateinamens hinterlegt werden >>
  03 PRNTDOC-dname-len PIC 9(04) COMP.
* length of the document-name
  88 PRNTDOC-dname-len-std VALUE 54.
  03 PRNTDOC-dname-ptr PIC S9(09) COMP.
* name of the file
  88 PRNTDOC-dname-ptr-none VALUE -1.

02 PRNTDOC-eamn OCCURS 16 TIMES PIC 9(05) BINARY.
<< Definition einer eindimensionalen Tabelle mit 16 Elementen zur Auf- >>
<< nahme von Dateinummern, wenn EAM-Dateien ausgedruckt werden. >>
* eam-file-number (range 0..65535)
  88 PRNTDOC-eamn-std VALUE ZERO.

02 PRNTDOC-sysnn OCCURS 16 TIMES PIC 9(04) COMP.
<< Definition einer eindimensionalen Tabelle mit 16 Elementen zur Auf- >>
<< nahme von Dateinummern, wenn SYSLST-Dateien ausgedruckt werden. >>
  88 PRNTDOC-sysnn-std VALUE ZERO.

```

**02 PRNTDOC-libelem.**

```

<< Datenstruktur, die den Namen, die Version und den Typ eines          >>
<< auszudruckenden Elements aus einer PLAM-Bibliothek zusammenfasst. >>
<< Siehe auch Beschreibung des Operanden "LIBELEM" auf Seite 79.      >>
03 PRNTDOC-element                PIC X(80).
*   element name
    88 PRNTDOC-element-dummy                VALUE .. ..
03 PRNTDOC-version                PIC X(40).
*   element version
    88 PRNTDOC-element-highest              VALUE .. ..
03 PRNTDOC-type                   PIC X(12).
*   element type

```

**02 PRNTDOC-sectrec.**

```

<< Datenstruktur, die eine Zeichenfolge in den Datensätzen, die als  >>
<< Abschnittsmarke fungieren soll und die Position der Zeichenfolge >>
<< zusammenfasst.                                                    >>
<< Siehe auch Beschreibung des Operanden "SECTREC" auf Seite 102.   >>
*   section records
    03 PRNTDOC-sect-id                PIC X(60).
    03 PRNTDOC-sect-pos                PIC S9(04) COMP.
    03 PRNTDOC-sect-len                PIC X(01).
    03 PRNTDOC-sect-ind                PIC X(01).

```

**02 PRNTDOC-firsrec.**

```

<< Datenstruktur, die alle Angaben für einen Anfangsdatsatz zusammen- >>
<< fasst, wenn von der Eingabedatei nur eine bestimmte Anzahl von Daten- >>
<< sätzen verarbeitet werden soll. In der Datenstruktur "-lastrec"   >>
<< können die Angaben für den Enddatsatz eingestellt werden.        >>
<< In der hier beschriebenen Datenstruktur werden die folgenden Angaben >>
<< zusammengefasst: Nummer eines Datensatzes, ab dem die Datei verarbei- >>
<< tet werden soll; Zeichenfolge, deren Auftreten einen Anfangsdaten- >>
<< satz repräsentiert soll; Position innerhalb des Datensatzes, ab der >>
<< die Zeichenfolge beginnt; Anzahl des Auftretens der Zeichenfolge;  >>
<< Länge der Zeichenfolge; Kennzeichen, ob die Zeichenfolge im       >>
<< Zeichenformat oder hexadezimal angegeben wird;                    >>
<< Siehe auch Beschreibung des Operanden "FIRSREC" auf Seite 64.     >>
*   beginning part
    03 PRNTDOC-stpos                PIC S9(09) COMP.
    03 PRNTDOC-ststrid                PIC X(60).
    03 PRNTDOC-ststrpos                PIC S9(04) COMP.
    03 PRNTDOC-ststrocc                PIC S9(04) COMP.
    03 PRNTDOC-ststrlen                PIC X(01).
    03 PRNTDOC-filler6                PIC X(01).
    03 PRNTDOC-strid-ind.
        04 PRNTDOC-st-char                PIC X(01).
            88 PRNTDOC-st-char-char                VALUE SYS-X80.
            88 PRNTDOC-st-char-hexa                VALUE SYS-X00.
    04 PRNTDOC-ststrid-free                PIC X(01).

```

**02 PRNTDOC-lastrec.**

<< Datenstruktur, die alle Angaben für einen Enddatensatz zusammen- >>  
 << fasst, wenn von der Eingabedatei nur eine bestimmte Anzahl von Daten- >>  
 << sätzen verarbeitet werden soll. In der Datenstruktur "-firsrec" >>  
 << können die Angaben für den Anfangsdatensatz eingestellt werden. >>  
 << In der hier beschriebenen Datenstruktur werden die folgenden Angaben >>  
 << zusammengefasst: Nummer eines Datensatzes, bis zu dem die Datei ver- >>  
 << arbeitet werden soll; Zeichenfolge, deren Auftreten einen Enddaten- >>  
 << satz repräsentiert soll; Position innerhalb des Datensatzes, ab der >>  
 << die Zeichenfolge beginnt; Anzahl des Auftretens der Zeichenfolge; >>  
 << Länge der Zeichenfolge; Kennzeichen, ob die Zeichenfolge im >>  
 << Zeichenformat oder hexadezimal angegeben wird; >>  
 << Siehe auch Beschreibung des Operanden "LASTREC" auf [Seite 76](#). >>

```
*      end part
03 PRNTDOC-enpos          PIC S9(09) COMP.
03 PRNTDOC-enstrid       PIC X(60).
03 PRNTDOC-enstropos    PIC S9(04) COMP.
03 PRNTDOC-enstrocc     PIC S9(04) COMP.
03 PRNTDOC-enstrlen     PIC X(01).
03 PRNTDOC-filler7      PIC X(01).
03 PRNTDOC-strid-ind.
  04 PRNTDOC-en-char     PIC X(01).
    88 PRNTDOC-en-char-char      VALUE SYS-X80.
    88 PRNTDOC-en-char-hexa     VALUE SYS-X00.
  04 PRNTDOC-enstrid-free    PIC X(01).
```

**02 PRNTDOC-recpart.**

<< Datenstruktur, in der die erste und letzte Spalte aller ausgewählten >>  
 << Datensätze zusammenfasst sind, falls nur ein bestimmter Teil jedes >>  
 << Satzes zu verarbeiten ist. Sie auch Operand "RECPART" auf [Seite 98](#). >>

```
*      record-part
03 PRNTDOC-first-ch      PIC S9(04) COMP.
03 PRNTDOC-last-ch      PIC S9(04) COMP.
  88 PRNTDOC-last-ch-std      VALUE ZERO.
```

**02 PRNTDOC-linesp.**

<< Datenstruktur, in der die Position des Vorschubsteuerzeichens und >>  
 << die Anzahl der Zeilenvorschübe bzw. die Art der Steuerzeichenauswer- >>  
 << tung zusammengefasst sind. Die Art der Steuerzeichenauswertung wird >>  
 << über eine Ganzzahl gesteuert, die jeweils eine bestimmte Steuerzei- >>  
 << chenauswertung repräsentiert. Jedem Bedingungsnamen in diesem >>  
 << strukturierten Datenfeld ist ein ganzzahliger Wert zugeordnet. >>  
 << Zur Beziehung zwischen Wert und Dateityp siehe auch die Beschreibung >>  
 << des Operanden "LINESP" auf [Seite 82](#). >>

```
*      line feeds after
03 PRNTDOC-cc-pos       PIC S9(04) COMP.
03 PRNTDOC-spacing     PIC X(01).
  88 PRNTDOC-space-1    VALUE SYS-X01.
  88 PRNTDOC-space-2    VALUE SYS-X02.
  88 PRNTDOC-space-3    VALUE SYS-X04.
```

```

      88 PRNTDOC-space-by-ebc-control          VALUE SYS-X08.
      88 PRNTDOC-space-by-asa-control         VALUE SYS-X10.
      88 PRNTDOC-space-by-ibm-control        VALUE SYS-X20.
      88 PRNTDOC-space-no                    VALUE SYS-X90.
      88 PRNTDOC-space-std                   VALUE SYS-X00.

```

```

03 PRNTDOC-filler4                          PIC X(01).

```

## 02 PRNTDOC-docform PIC X(01).

```

<< Datenstruktur, in der die Art des Dokumentinhaltes zusammengefasst ist. >>
<< Die Art des Dokumentinhaltes wird über eine Ganzzahl >>
<< gesteuert, die jeweils einen bestimmten Dokumentinhalt repräsentiert. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>
<< ganzzahliger Wert zugeordnet. >>
<< Zur Beziehung zwischen Wert und der Art des Dokumentinhaltes siehe >>
<< auch die Beschreibung des Operanden "DOCFORM" auf Seite 60. >>

```

```

*      docform-set
      88 PRNTDOC-docform-text                VALUE SYS-X00.
      88 PRNTDOC-docform-page-format        VALUE SYS-X01.
      88 PRNTDOC-docform-spec-format        VALUE SYS-X02.

```

## 02 PRNTDOC-contmod.

```

<< Datenstruktur, in der die Art der Steuerzeichenauswertung zusammenge- >>
<< fasst ist. Die Art der Steuerzeichenauswertung wird über eine Ganzzahl >>
<< gesteuert, die jeweils eine bestimmten Steuerzeichenauswertung reprä- >>
<< sentiert. Jedem Bedingungsnamen in diesem strukturierten Datenfeld >>
<< ist ein ganzzahliger Wert zugeordnet. Zur Beziehung zwischen Wert und >>
<< der Interpretation der Steuerzeichen siehe auch die Beschreibung >>
<< des Operanden "CONTMOD=mode" auf Seite 53. >>

```

```

*      control mode
03 PRNTDOC-ctlmode                          PIC X(01).
*      control-mode-set
      88 PRNTDOC-ctl-mode-std                VALUE SYS-X01.
      88 PRNTDOC-ctl-mode-physical          VALUE SYS-X02.
      88 PRNTDOC-ctl-mode-page-mode        VALUE SYS-X04.
      88 PRNTDOC-ctl-mode-logical          VALUE SYS-X08.
      88 PRNTDOC-ctl-mode-line-mode        VALUE SYS-X10.
      88 PRNTDOC-ctl-mode-apa              VALUE SYS-X40.
03 PRNTDOC-pcc                              PIC X(01).
*      pcc-set
      88 PRNTDOC-pcc-yes                    VALUE SYS-X00.
      88 PRNTDOC-pcc-no                     VALUE SYS-X01.
      88 PRNTDOC-pcc-dummy                  VALUE SYS-X02.
03 PRNTDOC-ctltype                          PIC X(01).
*      control-type-set
      88 PRNTDOC-ctl-type-dummy             VALUE SYS-X00.
      88 PRNTDOC-ctl-type-hp                VALUE SYS-X01.
      88 PRNTDOC-ctl-type-compatible        VALUE SYS-X02.

```

```

02 PRNTDOC-family                                PIC X(01).
<< Datenstruktur, in der die Vergabe einer gemeinsamen TSN zusammenge- >>
<< fasst ist. Die Vergabe einer gemeinsamen TSN wird über eine Ganzzahl >>
<< gesteuert, die jeweils eine bestimmte Option repräsentiert. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld >>
<< ist ein ganzzahliger Wert zugeordnet. Zur Beziehung zwischen Wert und >>
<< der Family-Funktion siehe auch die Beschreibung des Operanden "FAMILY">>
<< auf Seite 53. >>
*   family-set
    88 PRNTDOC-family-std                VALUE SYS-X01.
    88 PRNTDOC-family-yes                VALUE SYS-X02.
    88 PRNTDOC-family-no                VALUE SYS-X03.

02 PRNTDOC-checkp                                PIC X(01).
<< Datenstruktur, in der die Checkpoint-Verarbeitung zusammengefasst ist. >>
<< Die Checkpoint-Verarbeitung wird über eine Ganzzahl gesteuert, die >>
<< jeweils eine bestimmte Option repräsentiert. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld >>
<< ist ein ganzzahliger Wert zugeordnet. Zur Beziehung zwischen Wert und >>
<< dem Wiederanlauf siehe auch die Beschreibung des Operanden "CHECKP" >>
<< auf Seite 52. >>
*   checkp-set
    88 PRNTDOC-checkp-on-pages           VALUE SYS-X00.
    88 PRNTDOC-checkp-on-sect-recs      VALUE SYS-X01.
    02 PRNTDOC-doc-number                PIC 9(04) COMP.
    02 PRNTDOC-stproc-val                PIC S9(09) COMP.
    02 PRNTDOC-stproc                    PIC X(01).
*   stproc-val and stproc are only for internal usage

02 PRNTDOC-lockf                                PIC X(01).
<< Datenstruktur, die den Dateischutz während der Verarbeitung zusammen- >>
<< fasst. Der Dateischutz wird über eine Ganzzahl gesteuert, die jeweils >>
<< eine bestimmte Option repräsentiert. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld >>
<< ist ein ganzzahliger Wert zugeordnet. Zur Beziehung zwischen Wert und >>
<< dem Dateischutz siehe auch die Beschreibung des Operanden "LOCKF" >>
<< auf Seite 83. >>
*   lock while job is waiting
*   lockf-set
    88 PRNTDOC-lockf-std                 VALUE SYS-X00.
    88 PRNTDOC-lockf-no                 VALUE SYS-X01.
    88 PRNTDOC-lockf-yes                 VALUE SYS-X02.

02 PRNTDOC-delf                                  PIC X(01).
<< Datenstruktur, die das Löschen der Datei zusammenfasst. >>
<< Das Löschen der Datei wird über eine Ganzzahl gesteuert, die jeweils >>
<< eine bestimmte Option repräsentiert. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>
<< ganzzahliger Wert zugeordnet. Zur Beziehung zwischen Wert und dem >>
<< Löschen der Datei siehe auch die Beschreibung des Operanden "DELf" >>
<< auf Seite 58. >>

```



```

*      delete (destroy) document
*      delf-set
      88 PRNTDOC-delf-yes          VALUE SYS-X01.
      88 PRNTDOC-delf-destroy     VALUE SYS-X02.
      88 PRNTDOC-delf-no         VALUE SYS-X03.
02 PRNTDOC-addcop                PIC X(01).
<< Datenfeld zur Aufnahme eines Wertes für zusätzliche Ausdrücke der >>
<< Datei. >>
<< Siehe auch die Beschreibung des Operanden "ADDCOP" auf Seite 51. >>
*      additional copies
02 PRNTDOC-jvpassi              PIC S9(09) COMP.
<< Datenfeld zur Aufnahme eines (numerischen) Kennwortes, mit dem die >>
<< die auftragsüberwachende Jobvariable geschützt wird. >>
<< Siehe auch die Beschreibung des Operanden "JVPASSW" auf Seite 76. >>
*      password on monitor job-variable (numeric)
02 PRNTDOC-jvpassw             REDEFINES PRNTDOC-jvpassi
                                PIC X(04).
<< Datenfeld zur Aufnahme eines (alphanum.) Kennwortes , mit dem die >>
<< die auftragsüberwachende Jobvariable geschützt wird. >>
<< Siehe auch die Beschreibung des Operanden "JVPASSW" auf Seite 76. >>
*      password on monitor job-variable
02 PRNTDOC-monjv               PIC X(54).
<< Datenfeld zur Aufnahme einer auftragsüberwachenden Jobvariablen. >>
<< Siehe auch die Beschreibung des Operanden "MONJV" auf Seite 85. >>
*      monitor job-variable
02 PRNTDOC-prjname            PIC X(08).
<< Datenfeld zur Aufnahme des Auftragsnamens. >>
<< Siehe auch die Beschreibung des Operanden "PRJNAME" auf Seite 93. >>
*      job-name of printjob
02 PRNTDOC-prjprio            PIC X(01).
<< Datenfeld zur Aufnahme des Jobpriorität. >>
<< Siehe auch die Beschreibung des Operanden "PRJPRI0" auf Seite 93. >>
*      priority
02 PRNTDOC-procadm.
<< Datenstruktur, in der die Attribute einer Benutzererkennung hinterlegt >>
<< sind, wenn der SPOOLOUT-Auftrag unter einer anderen als der Aufrufer- >>
<< Benutzererkennung verarbeitet werden soll. >>
<< Siehe auch Beschreibung des Operanden "PROCADM" auf Seite 95. >>
*      execute print-job under another user-id
      03 PRNTDOC-user             PIC X(08).
      03 PRNTDOC-account         PIC X(08).
      03 PRNTDOC-password       PIC X(08).
02 PRNTDOC-truncation        PIC X(01).
<< Datenstruktur, in der das Verhalten bei abgeschnittenen Zeilen hin- >>
<< terlegt wird. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>
<< ganzzahliger Wert zugeordnet. >>
<< Beziehung zwischen Wert und dem Verhalten bei abgeschnittenen Zeilen >>

```

<< siehe auch die Beschreibung des Operanden "TRUNC" auf [Seite 108.](#) >>

```
* truncation-type-set
    88 PRNTDOC-truncation-std          VALUE SYS-X00.
    88 PRNTDOC-truncation-delete-file  VALUE SYS-X01.
    88 PRNTDOC-truncation-keep-file    VALUE SYS-X02.
```

**02 PRNTDOC-outpart.**

<< Datenstruktur, in der die Untermenge aller logischen Druckseiten, >>  
 << auf die sich die Ausgabe beschränken soll, definiert werden kann. >>  
 << Die Datenstruktur umfasst eine Seiten- oder Zeilennummer, mit der die >>  
 << Ausgabe beginnen und eine Seiten- oder Zeilennummer, mit der die >>  
 << Ausgabe enden soll. Bzgl. der Dimension (Seiten- oder Zeilennummer) >>  
 << kann auf eine Konstante zurückgegriffen werden, die in der >>  
 << Bedingungsnamen-Reihe "layout-part-dim-set" aufgelistet ist. >>  
 << Siehe auch Beschreibung des Operanden "OUTPART" auf [Seite 86.](#) >>

```
* layout control (number of pages or lines)
    03 PRNTDOC-rec-from                PIC S9(09) COMP.
    03 PRNTDOC-rec-to                  PIC S9(09) COMP.
    03 PRNTDOC-dim                     PIC X(01).
```

```
* layout-part-dim-set
    88 PRNTDOC-layout-part-pages      VALUE SYS-X01.
    88 PRNTDOC-layout-part-lines      VALUE SYS-X02.
    03 PRNTDOC-filler5                PIC X(03).
    02 PRNTDOC-filler1                 PIC X(01).
    02 PRNTDOC-leftmarg                PIC X(01).
```

**02 PRNTDOC-leftmarg**

<< Datenfeld, in dem die Einrückung des des Ausgabebetextes vereinbart wird.>>  
 << Siehe auch die Beschreibung des Operanden "LEFTMAR" auf [Seite 78.](#) >>

```
* indentation for the output text
    88 PRNTDOC-leftmarg-std            VALUE SYS-XFF.
    02 PRNTDOC-leftmarg                PIC S9(04) COMP.
```

**02 PRNTDOC-linepp**

<< Datenfeld, in dem die Anzahl Zeilen pro Seite vereinbart wird. >>  
 << Siehe auch die Beschreibung des Operanden "LINEPP" auf [Seite 81.](#) >>

```
* lines per page (including header and blanks)
```

**02 PRNTDOC-headline** PIC X(01).

<< Datenstruktur, in der die Vereinbarung bzgl. der Überschriftszeile >>  
 << definiert wird. >>  
 << Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>  
 << ganzzahliger Wert zugeordnet. >>  
 << Zur Beziehung zwischen Wert und der Überschriftszeile >>  
 << siehe auch die Beschreibung des Operanden "HEADLIN" auf [Seite 68.](#) >>

```
* date/first-record/page-number
    88 PRNTDOC-headline-no             VALUE SYS-X01.
    88 PRNTDOC-headline-std            VALUE SYS-X02.
    88 PRNTDOC-headline-record         VALUE SYS-X04.
    88 PRNTDOC-headline-date           VALUE SYS-X08.
    88 PRNTDOC-headline-date-record    VALUE SYS-X0C.
    88 PRNTDOC-headline-page           VALUE SYS-X10.
    88 PRNTDOC-headline-record-page    VALUE SYS-X14.
    88 PRNTDOC-headline-date-page      VALUE SYS-X18.
```

```

      88 PRNTDOC-headline-date-rec-page          VALUE SYS-X1C.
02 PRNTDOC-outform                          PIC X(01).
<< Datenstruktur, in der die Vereinbarung bzgl. der Ausgabe der Zeichen >>
<< definiert wird. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>
<< ganzzahliger Wert zugeordnet. >>
<< Zur Beziehung zwischen Wert und der Ausgabe der Zeichen siehe auch >>
<< die Beschreibung des Operanden "OUTFORM" auf Seite 85. >>
*      output-format
*      outform-set
      88 PRNTDOC-outform-character              VALUE SYS-X01.
      88 PRNTDOC-outform-hexadecimal           VALUE SYS-X02.
02 PRNTDOC-twoside                          PIC X(01).
<< Datenstruktur, in der die Vereinbarung bzgl. des doppelseitigen >>
<< Ausdrucks definiert wird. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>
<< ganzzahliger Wert zugeordnet. >>
<< Zur Beziehung zwischen Wert und des doppelseitigen Ausdrucks siehe >>
<< auch die Beschreibung des Operanden "TWOSIDE" auf Seite 109. >>
*      print on one or both sides of the paper
*      twoside-set
      88 PRNTDOC-twoside-std                    VALUE SYS-X00.
      88 PRNTDOC-twoside-yes                    VALUE SYS-X01.
      88 PRNTDOC-twoside-tumble                 VALUE SYS-X02.
      88 PRNTDOC-twoside-no                     VALUE SYS-X04.
02 PRNTDOC-rotation                        PIC X(01).
<< Datenstruktur, in der die Vereinbarung bzgl. einer Seitendrehung >>
<< des Ausdrucks definiert wird. >>
<< Jedem Bedingungsnamen in diesem strukturierten Datenfeld ist ein >>
<< ganzzahliger Wert zugeordnet. >>
<< Zur Beziehung zwischen Wert und der Seitendrehung siehe auch die >>
<< Beschreibung des Operanden "ROT" auf Seite 100. >>
*      rotate the text when printing it
*      rotation-set
      88 PRNTDOC-rotation-no                     VALUE „N“.
      88 PRNTDOC-rotation-0                       VALUE „0“.
      88 PRNTDOC-rotation-90                      VALUE „1“.
      88 PRNTDOC-rotation-180                     VALUE „2“.
      88 PRNTDOC-rotation-270                     VALUE „3“.
      88 PRNTDOC-rotation-0-180                   VALUE „4“.
      88 PRNTDOC-rotation-90-270                  VALUE „5“.
      88 PRNTDOC-rotation-180-0                   VALUE „6“.
      88 PRNTDOC-rotation-270-90                  VALUE „7“.
      88 PRNTDOC-rot-by-control-codes              VALUE „Y“.
02 PRNTDOC-intray                          PIC X(01).
<< Datenfeld, in dem das Papiereingabefach für LP65-Drucker vereinbart >>
<< wird. >>
<< Siehe auch die Beschreibung des Operanden "INTRAY" auf Seite 70. >>

```

```

*      input-tray (only LP65 and RSO-printers)
02 PRNTDOC-outtray          PIC X(01).
<< Datenfeld, in dem das Papierausabefach für LP65-Drucker vereinbart >>
<< wird. >>
<< Siehe auch die Beschreibung des Operanden "OUTTRAY" auf Seite 88. >>
*      output-tray (only LP65 and RSO-printers)
02 PRNTDOC-pagecop        PIC S9(04) COMP.
<< Datenfeld, in dem die Anzahl Kopien für jede Seite vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "PAGECOP" auf Seite 90. >>
*      copies of each printed page
      88 PRNTDOC-pagecop-std          VALUE -1.
02 PRNTDOC-covpage.
<< Datenstruktur, in der die Angaben für Exit-Routinen, die das >>
<< Drucken von Deck- und Schlussblätter betreffen, zusammengefasst sind. >>
<< Siehe auch Beschreibung des Operanden "COVPAGE" auf Seite 57. >>
*      cover-page
      03 PRNTDOC-headtext          PIC X(32).
      03 PRNTDOC-headexit         PIC S9(09) COMP.
      88 PRNTDOC-headexit-no      VALUE -1.
      03 PRNTDOC-traiaexit        PIC S9(09) COMP.
      88 PRNTDOC-traiaexit-no     VALUE -1.
02 PRNTDOC-form          PIC X(06).
<< Datenfeld, in dem der Name des zu verwendenden Formulars vereinbart >>
<< wird. >>
<< Siehe auch die Beschreibung des Operanden "FORM" auf Seite 66. >>
02 PRNTDOC-loop        PIC X(03).
<< Datenfeld, in dem der Name des zu verwendenden Loops vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "LOOP" auf Seite 84. >>
02 PRNTDOC-rotloop     PIC X(03).
<< Datenfeld, in dem der Name des zu verwendenden Rotation-Loops verein- >>
<< bart wird. >>
<< Siehe auch die Beschreibung des Operanden "ROTLLOOP" auf Seite 101. >>
02 PRNTDOC-charset OCCURS 16 TIMES PIC X(03).
<< Definition einer eindimensionalen Tabelle mit 16 Elementen zur Auf- >>
<< nahme von Fonts, die für den Ausdruck verwendet werden. >>
<< Siehe auch die Beschreibung des Operanden "CHARSET" auf Seite 51. >>
*      one or more character-sets used for printing
02 PRNTDOC-poolnam      PIC X(04).
<< Datenfeld, in dem der Name des zu verwendenden Font-Pools vereinbart >>
<< wird. >>
<< Siehe auch die Beschreibung des Operanden "POOLNAM" auf Seite 92. >>
      88 PRNTDOC-poolnam-none      VALUE SPACES.
02 PRNTDOC-poolind     PIC X(01).
<< Datenfeld, in dem die Nummer eines Fonts aus dem Font-Pool vereinbart >>
<< wird. >>
<< Siehe auch die Beschreibung des Operanden "POOLIND" auf Seite 91. >>

```

```

02 PRNTDOC-efo PIC X(02).
<< Datenfeld, in dem der Name eines Filmdias vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "EFO" auf Seite 62. >>
* film formovl for printing

02 PRNTDOC-overlay PIC X(02).
<< Datenstruktur, in der die Angaben für EFO-Datendias für Drucker vom >>
<< Typ LP65 zusammengefasst sind. >>
<< Siehe auch Beschreibung des Operanden "OVERLAY" auf Seite 89. >>
02 PRNTDOC-overlay-par REDEFINES PRNTDOC-overlay.
03 PRNTDOC-face PIC X(01).
88 PRNTDOC-face-std VALUE SYS-XFF.
88 PRNTDOC-face-none VALUE SYS-X00.
03 PRNTDOC-reverse PIC X(01).
88 PRNTDOC-reverse-std VALUE SYS-XFF.
88 PRNTDOC-reverse-none VALUE SYS-X00.

02 PRNTDOC-fob PIC X(04).
<< Datenfeld, in dem der Name eines FOB-Datendias vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "FOB" auf Seite 66. >>
* Forms Overlay Buffer
02 PRNTDOC-filler2 PIC X(03).
* slack bytes

02 PRNTDOC-pagepcl PIC S9(09) COMP.
<< Datenfeld, in dem die Nummer der PCL-Datei vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "PAGEPCL" auf Seite 91. >>

02 PRNTDOC-userres PIC X(44).
<< Datenfeld, in dem der Name der Ressourcen-Datei vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "USERRES" auf Seite 110. >>
* user resources file

02 PRNTDOC-transtab.
<< Datenstruktur, in der die Angaben zur Verarbeitung einer Code-Umsetz- >>
<< tabelle (Name der Tabelle, Datei) zusammengefasst sind. >>
<< Siehe auch Beschreibung des Operanden "TRANTAB" auf Seite 107. >>
* translation table
03 PRNTDOC-name PIC X(08).
03 PRNTDOC-file PIC X(44).

02 PRNTDOC-prname PIC X(08).
<< Datenfeld, in dem der Name des Zieldruckers vereinbart wird. >>
<< Siehe auch die Beschreibung des Operanden "PRNAME" auf Seite 94. >>
* printer or printers pool

02 PRNTDOC-prtype PIC X(01).
<< Datenfeld, in dem der Druckertyp vereinbart wird. >>
<< Welchen Druckertypen angegeben werden können, ist der Datenstruktur >>
<< auf Seite 153 zu entnehmen. >>
<< Siehe auch die Beschreibung des Operanden "PRTYPE" auf Seite 96. >>
* combination of printers
88 PRNTDOC-prtype-any VALUE SYS-X00.
02 PRNTDOC-filler3 PIC X(03).
02 PRNTDOC-tsn PIC X(04).

```

```
*      Output-field: TSN obtained from the system
02 PRNTDOC-cluster            PIC X(08).
*      Only for compatibility with MODJPRT
```

**02 PRNTDOC-specified.**

```
<< Datenstruktur, in der durch Setzen des Bit-Indikators der Zugriff auf >>
<< eine Menge an möglichen Optionen (= Operanden des Assembler-Makros) >>
<< ermöglicht wird. Die Operanden sind jeweils zu Blöcken (specified1 bis >>
<< specified7) zusammengefasst und in der Datenstruktur "specified-set" auf >>
<< Seite 154) aufgelistet. >>
```

```
*      bits indicating which operands are given
03 PRNTDOC-specified1       PIC X(01).
03 PRNTDOC-specified2       PIC X(01).
03 PRNTDOC-specified3       PIC X(01).
03 PRNTDOC-specified4       PIC X(01).
03 PRNTDOC-specified5       PIC X(01).
03 PRNTDOC-specified6       PIC X(01).
03 PRNTDOC-specified7       PIC X(01).
03 PRNTDOC-specified8       PIC X(01).
```

```
-----*
*      Initialization of PRNTDOC-pl.                                     *
*-----*
```

**01 PRNTDOC-I-pl.**

```
<< Initialisieren der Parameter-Liste                                >>
02 PRNTDOC-I-hdr.
03 PRNTDOC-I-iffid.
04 PRNTDOC-I-unit           PIC S9(04) COMP
                             VALUE 46.
04 PRNTDOC-I-function       PIC X(01)
                             VALUE SYS-X05.
04 PRNTDOC-I-version        PIC X(01)
                             VALUE SYS-X01.
03 PRNTDOC-I-rc-nbr         PIC S9(09) COMP
                             VALUE -1.
02 PRNTDOC-I-rso-pl         PIC S9(09) COMP
                             VALUE -1.
02 PRNTDOC-I-sps-pl         PIC S9(09) COMP
                             VALUE -1.
02 PRNTDOC-I-dpri-pl        PIC S9(09) COMP
                             VALUE -1.
02 PRNTDOC-I-dsem-pl        PIC S9(09) COMP
                             VALUE -1.
02 PRNTDOC-I-reserved-1     PIC S9(09) COMP
                             VALUE ZERO.
02 PRNTDOC-I-reserved-2     PIC S9(09) COMP
                             VALUE ZERO.
02 PRNTDOC-I-reserved-3     PIC S9(09) COMP
                             VALUE ZERO.
02 PRNTDOC-I-reserved-4     PIC S9(09) COMP
```

```

                                VALUE ZERO.
02 PRNTDOC-I-dtype              PIC 9(04) COMP
                                VALUE 1.
02 PRNTDOC-I-dname.
  03 PRNTDOC-I-dname-len        PIC 9(04) COMP
                                VALUE 54.
  03 PRNTDOC-I-dname-ptr       PIC S9(09) COMP
                                VALUE -1.
02 PRNTDOC-I-eamn OCCURS 16 TIMES PIC 9(05) BINARY
                                VALUE ZERO.
02 PRNTDOC-I-sysnn OCCURS 16 TIMES PIC 9(04) COMP
                                VALUE ZERO.
02 PRNTDOC-I-libelem.
  03 PRNTDOC-I-element         PIC X(80)
                                VALUE SPACES.
  03 PRNTDOC-I-version         PIC X(40)
                                VALUE SPACES.
  03 PRNTDOC-I-type            PIC X(12)
                                VALUE SPACES.
02 PRNTDOC-I-sectrec.
  03 PRNTDOC-I-sect-id         PIC X(60)
                                VALUE SPACES.
  03 PRNTDOC-I-sect-pos       PIC S9(04) COMP
                                VALUE ZERO.
  03 PRNTDOC-I-sect-len       PIC X(01)
                                VALUE SYS-X00.
  03 PRNTDOC-I-sect-ind       PIC X(01)
                                VALUE SYS-X00.
02 PRNTDOC-I-firsrec.
  03 PRNTDOC-I-stpos          PIC S9(09) COMP
                                VALUE 1.
  03 PRNTDOC-I-ststrid        PIC X(60)
                                VALUE SPACES.
  03 PRNTDOC-I-ststrpos       PIC S9(04) COMP
                                VALUE ZERO.
  03 PRNTDOC-I-ststrocc       PIC S9(04) COMP
                                VALUE 1.
  03 PRNTDOC-I-ststrlen       PIC X(01)
                                VALUE SYS-X00.
  03 PRNTDOC-I-filler6       PIC X(01)
                                VALUE SPACE.
  03 PRNTDOC-I-strid-ind.
    04 PRNTDOC-I-st-char      PIC X(01)
                                VALUE SYS-X00.
    04 PRNTDOC-I-ststrid-free PIC X(01)
                                VALUE LOW-VALUE.
02 PRNTDOC-I-lastrec.
  03 PRNTDOC-I-enpos.

```

04 FILLER	PIC X(01) VALUE SYS-X7F.
04 FILLER	PIC X(03) VALUE HIGH-VALUES.
03 PRNTDOC-I-enstrid	PIC X(60) VALUE SPACES.
03 PRNTDOC-I-enstrpos	PIC S9(04) COMP VALUE ZERO.
03 PRNTDOC-I-enstrocc	PIC S9(04) COMP VALUE 1.
03 PRNTDOC-I-enstrlen	PIC X(01) VALUE SYS-X00.
03 PRNTDOC-I-filler7	PIC X(01) VALUE SPACE.
03 PRNTDOC-I-strid-ind.	
04 PRNTDOC-I-en-char	PIC X(01) VALUE SYS-X00.
04 PRNTDOC-I-enstrid-free	PIC X(01) VALUE LOW-VALUE.
02 PRNTDOC-I-recpart.	
03 PRNTDOC-I-first-ch	PIC S9(04) COMP VALUE 1.
03 PRNTDOC-I-last-ch	PIC S9(04) COMP VALUE ZERO.
02 PRNTDOC-I-linesp.	
03 PRNTDOC-I-cc-pos	PIC S9(04) COMP VALUE 0.
03 PRNTDOC-I-spacing	PIC X(01) VALUE SYS-X00.
03 PRNTDOC-I-filler4	PIC X(01) VALUE SPACE.
02 PRNTDOC-I-docform	PIC X(01) VALUE SYS-X00.
02 PRNTDOC-I-contmod.	
03 PRNTDOC-I-ctlmode	PIC X(01) VALUE SYS-X00.
03 PRNTDOC-I-pcc	PIC X(01) VALUE SYS-X02.
03 PRNTDOC-I-ctltype	PIC X(01) VALUE SYS-X00.
02 PRNTDOC-I-family	PIC X(01) VALUE SYS-X01.
02 PRNTDOC-I-checkp	PIC X(01) VALUE SYS-X00.
02 PRNTDOC-I-doc-number	PIC 9(04) COMP VALUE ZERO.
02 PRNTDOC-I-stproc-val	PIC S9(09) COMP VALUE ZERO.



```

02 PRNTDOC-I-stproc          PIC X(01)
                             VALUE SYS-X04.
02 PRNTDOC-I-lockf         PIC X(01)
                             VALUE SYS-X00.
02 PRNTDOC-I-de1f         PIC X(01)
                             VALUE SYS-X03.
02 PRNTDOC-I-addcop       PIC X(01)
                             VALUE LOW-VALUE.
02 PRNTDOC-I-jvpassi     PIC S9(09) COMP
                             VALUE ZERO.
02 PRNTDOC-I-monjv       PIC X(54)
                             VALUE SPACES.
02 PRNTDOC-I-prjname     PIC X(08)
                             VALUE SPACES.
02 PRNTDOC-I-prjprio     PIC X(01)
                             VALUE LOW-VALUE.
02 PRNTDOC-I-procadm.
  03 PRNTDOC-I-user       PIC X(08)
                             VALUE SPACES.
    03 PRNTDOC-I-account  PIC X(08)
                             VALUE SPACES.
    03 PRNTDOC-I-password PIC X(08)
                             VALUE SPACES.
02 PRNTDOC-I-truncation  PIC X(01)
                             VALUE SYS-X00.
02 PRNTDOC-I-outpart.
  03 PRNTDOC-I-rec-from   PIC S9(09) COMP
                             VALUE ZERO.
    03 PRNTDOC-I-rec-to   PIC S9(09) COMP
                             VALUE ZERO.
    03 PRNTDOC-I-dim      PIC X(01)
                             VALUE SYS-X01.
    03 PRNTDOC-I-filler5  PIC X(03)
                             VALUE SPACES.
02 PRNTDOC-I-filler1     PIC X(01)
                             VALUE SPACE.
02 PRNTDOC-I-leftmarg    PIC X(01)
                             VALUE SYS-XFF.
02 PRNTDOC-I-linepp      PIC S9(04) COMP
                             VALUE ZERO.
02 PRNTDOC-I-headline    PIC X(01)
                             VALUE SYS-X01.
02 PRNTDOC-I-outform     PIC X(01)
                             VALUE SYS-X01.
02 PRNTDOC-I-twoside     PIC X(01)
                             VALUE SYS-X00.
02 PRNTDOC-I-rotation    PIC X(01)
                             VALUE „N“.

```

```

02 PRNTDOC-I-intray          PIC X(01)
                             VALUE SYS-X00.
02 PRNTDOC-I-outtray        PIC X(01)
                             VALUE SYS-X00.
02 PRNTDOC-I-pagecop        PIC S9(04) COMP
                             VALUE -1.
02 PRNTDOC-I-covpage.
  03 PRNTDOC-I-headtext     PIC X(32)
                             VALUE SPACES.
  03 PRNTDOC-I-headexit     PIC S9(09) COMP
                             VALUE -1.
  03 PRNTDOC-I-traiaexit    PIC S9(09) COMP
                             VALUE -1.
02 PRNTDOC-I-form           PIC X(06)
                             VALUE SPACES.
02 PRNTDOC-I-loop          PIC X(03)
                             VALUE SPACES.
02 PRNTDOC-I-rotloop        PIC X(03)
                             VALUE SPACES.
02 PRNTDOC-I-charset-all   PIC X(48)
                             VALUE SPACES.
02 PRNTDOC-I-poolnam        PIC X(04)
                             VALUE SPACES.
02 PRNTDOC-I-poolind        PIC X(01)
                             VALUE LOW-VALUE.
02 PRNTDOC-I-efo           PIC X(02)
                             VALUE SPACES.
02 PRNTDOC-I-ovlmode-face   PIC X(01)
                             VALUE SYS-XFF.
02 PRNTDOC-I-ovlmode-reverse PIC X(01)
                             VALUE SYS-XFF.
02 PRNTDOC-I-fob           PIC X(04)
                             VALUE SPACES.
02 PRNTDOC-I-filler2        PIC X(03)
                             VALUE SPACES.
02 PRNTDOC-I-pagepcl        PIC S9(09) COMP
                             VALUE ZERO.
02 PRNTDOC-I-userres        PIC X(44)
                             VALUE SPACES.
02 PRNTDOC-I-transtab.
  03 PRNTDOC-I-name         PIC X(08)
                             VALUE SPACES.
  03 PRNTDOC-I-file         PIC X(44)
                             VALUE SPACES.
02 PRNTDOC-I-prname         PIC X(08)
                             VALUE SPACES.
02 PRNTDOC-I-ind3.
  03 PRNTDOC-I-prtype       PIC X(01)

```

```
                                VALUE LOW-VALUE.
02 PRNTDOC-I-filler3           PIC X(03)
                                VALUE SPACES.
02 PRNTDOC-I-tsn              PIC X(04)
                                VALUE SPACES.
02 PRNTDOC-I-cluster          PIC X(08)
                                VALUE SPACES.
02 PRNTDOC-I-specified        PIC X(08)
                                VALUE LOW-VALUES.
02 PRNTDOC-I-program.
  03 PRNTDOC-I-name            PIC X(08)
                                VALUE SPACES.
    03 PRNTDOC-I-string        PIC X(32)
                                VALUE SPACES.
02 PRNTDOC-I-format-name      PIC X(63)
                                VALUE SPACES.
02 PRNTDOC-I-modres           PIC S9(02) COMP
                                VALUE ZERO.
02 PRNTDOC-I-specified9       PIC X(01)
                                VALUE LOW-VALUE.
02 PRNTDOC-I-top-offset       PIC S9(04) COMP
                                VALUE -1.
02 PRNTDOC-I-left-offset      PIC S9(04) COMP
                                VALUE -1.
02 PRNTDOC-I-scheddate        PIC X(10)
                                VALUE "0000-00-00".
02 PRNTDOC-I-schedtime        PIC X(05)
                                VALUE "99:99".
02 PRNTDOC-I-unused           PIC X(11)
                                VALUE LOW-VALUES.
```

*Beispiel*

```

IDENTIFICATION DIVISION.
*-----*
*
program-id.
*-----*
*
        cobf5001.
/
ENVIRONMENT DIVISION.
*-----*
*
CONFIGURATION SECTION.
*-----*
*
SPECIAL-NAMES.
*-----*
*
        TERMINAL IS v-terminal,
        SYMBOLIC CHARACTERS
        COPY esmhexay. .
/
DATA DIVISION.
*-----*
*
WORKING-STORAGE SECTION.
*-----*
*
01 file-names.
   02 file-name-1      PIC X(54) VALUE „$TSOS.HELGA“.
01 hexa-chars         PIC X(16) VALUE „0123456789ABCDEF“.
01 maincode-edit.
   02 maincode-dec     PIC S9(9) COMP.
   02 FILLER           REDEFINES maincode-dec.
       03 maincode-byte PIC X(01) OCCURS 4 TIMES.
   02 maincode-hex     PIC X(08).
   02 FILLER           REDEFINES maincode-hex.
       03 maincode-char2 OCCURS 4 TIMES.
       04 maincode-char  PIC X(01) OCCURS 2 TIMES.
01 work-fields.
   02 work-counters.
       03 i            PIC S9(4) COMP.
       03 work-hw      PIC S9(4) COMP.
       03 FILLER       REDEFINES work-hw.
           04 work-hw-1 PIC X(01).
           04 work-hw-2 PIC X(01).

```

```

/
  COPY prntdocy SUPPRESS.
/
  PROCEDURE DIVISION.
*-----*
*
  s-main SECTION.
*-----*
*
  p-main.
*-----*
*
  PERFORM s-test001.
*
  p-exit.
*-----*
*
  STOP RUN.
/-----*
*
* TEST 001 : all default values
*
*-----*
*
  s-test001 SECTION.
*-----*
*
  p-test001-strt.
*-----*
*
  DISPLAY „TEST1 (DEF) OF COBF5001“ UPON v-terminal.
  MOVE prntdoc-i-pl TO prntdoc-pl.
  DISPLAY „File-name“ UPON v-terminal,
  ACCEPT file-name-1 FROM v-terminal.
*
  p-test001-init.
*-----*
*
  CALL „SCPADDR“ USING prntdoc-dname-ptr, file-name-1.
* COMPUTE prntdoc-dname-len = FUNCTION LENGTH(file-name-1),
  SET prntdoc-specified1-dname TO TRUE,
  CALL „SCPSETB1“ USING prntdoc-specified1,
  prntdoc-specified1-set.
*
  p-test001-call.
*-----*
*
  CALL „PRNTDOC“ USING prntdoc-pl.

```

```

*
p-test001-retc.
*-----*
*
    IF esmfhdr-rc-nbr IN prntdoc-p1 = ZERO
    THEN
        DISPLAY „MAINCODE = X'00000000“ UPON v-terminal,
        DISPLAY „TSN      = „, prntdoc-tsn UPON v-terminal,
    ELSE
        MOVE esmfhdr-rc-nbr IN prntdoc-p1 TO maincode-dec,
        PERFORM s-edit-maincode,
        DISPLAY „MAINCODE = X'“, maincode-hex, „‘“ UPON v-terminal,
    END-IF.
*
p-test001-exit.
*-----*
*
    EXIT.
/
s-edit-maincode SECTION.
*-----*
*
p-edit-maincode-strt.
*-----*
*
    PERFORM WITH TEST AFTER          VARYING i FROM 1 BY 1
    UNTIL i > FUNCTION LENGTH(maincode-dec)
    MOVE ZERO                          TO work-hw,
    MOVE maincode-byte(i)              TO work-hw-2,
    MOVE hexa-chars(work-hw / 16 + 1: 1)
                                        TO maincode-char(i, 1),
    MOVE hexa-chars(FUNCTION MOD(work-hw, 16) + 1: 1)
                                        TO maincode-char(i, 2),
    END-PERFORM.
*
p-edit-maincode-exit.
*-----*
*
    EXIT.
*****
*** ENDE *****
*****

```

## 2.3 Unterstützende Makros für PRNTDOC

Die folgenden Makros unterstützen den Einsatz des Makros PRNTDOC. Mit diesen Makros werden subsystemspezifische Parameter für das Makro PRNTDOC gesetzt.

„[PRNTRSO - RSO-Parameter für PRNTDOC-Makro festlegen](#)“ auf Seite 176

Mit diesem Makro werden Parameter für das Subsystem RSO gesetzt.

„[PRNTDPC - Dprint-Parameter für PRNTDOC-Makro definieren](#)“ auf Seite 182

Mit diesem Makro werden Parameter für das Subsystem Dprint gesetzt.

„[PRNTSPS - SPS-APA-Parameter für PRNTDOC-Makro definieren](#)“ auf Seite 187

Mit diesem Makro werden Parameter für das Subsystem SPS gesetzt.

## PRNTRSO - RSO-Parameter für PRNTDOC-Makro festlegen

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Der Makro PRNTRSO hinterlegt RSO-spezifische Werte für den Aufruf des Makros PRNTDOC in einer Parameterliste.

### Format

Operation	Operanden
PRNTRSO	<pre> ,VARIANT=<u>001</u> / &lt;c-string 3..3&gt;  ,CHRA=<u>*STD</u> / *ALL / *RESTRICTED /       &lt;var: enum-of _char_att_type_set:1&gt;  ,RDIR=<u>*STD</u> / *YES / *NO / &lt;var: enum-of _redir_type_set:1&gt;  ,TOPOFF=<u>*IGNORE</u> / &lt;integer -255..255&gt;  ,LEFTOFF=<u>*IGNORE</u> / &lt;integer -255..255&gt;  URL=(<i>urlptr,urlen,prtype</i>)   <i>urlptr</i>: &lt;var: pointer&gt; / *NONE   <i>urlen</i>: &lt;var: int 0..65535&gt;/ &lt;integer 1..1023&gt; / *NONE   <i>prtype</i>: &lt;var: enum-of:1 _PRTYPE_SET&gt; / *NONE /           *4850_PCL / *4830_PCL / *4818_PCL / *9000_EPSQ /           *9000_EPLQ / *9000_EPFX / *9000_PRO / *9000_PS /           *9046 / *9045_ANSI / *9015 / *4825_PCL / *4822_PCL /           *9001 / *9025 / *9013 / *9000 / *9001_31/ *9011 / *9012 /           *9022 / *9014 / *9021 / *9022_200 / *9000_PCL /           *4824_PCL / *9026_RENO / *9026_PCL / *4812 /           *9097 / *4011 / *4813 / *4821_PCL / *2030_PCL                     </pre>



## Operandenbeschreibung

### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **CHRA=\*STD / \*ALL / \*RESTRICTED / <var: enum-of \_char\_att\_type\_set:1>**

Legt fest, ob für den RSO-Auftrag alle oder nur bestimmte Zeichensatz-Eigenschaften berücksichtigt werden. Solche Eigenschaften sind z.B. Schriftart, Schönschrift (NLQ), Farbe u.s.w. (siehe Kommando oder SPSEVE-Anweisung SHOW-SPOOL-CHARACTER-SETS). Für Deck- und Schlussblätter ist dieser Operand nicht wirksam.

### **CHRA=\*STD / \*ALL**

Alle Eigenschaften der zum Ausdrucken benutzten Zeichensätze werden für den aktuellen RSO-Auftrag berücksichtigt.

### **CHRA=\*RESTRICTED**

Nur die drei Eigenschaften Schriftart (CHARACTER TYPE), Sprache (LANGUAGE) und NLQ (NEAR-LETTER-QUALITY) werden berücksichtigt.

### **CHRA=<var: enum-of \_char\_att\_type\_set:1>**

Die Vereinbarung bzgl. der zu verwendenden Zeichensatz-Eigenschaften wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und gewünschter Funktion:

0	*STD
1	*ALL
2	*RESTRICTED
siehe Beschreibung der Parameterliste ab <a href="#">Seite 181</a> .	

### **RDIR=\*STD / \*YES / \*NO / <var: enum-of \_redir\_type\_set:1>**

Gibt an, ob ein Geräteverwalter den RSO-Auftrag auf einen anderen Drucker umleiten darf. Das Umleiten von Aufträgen durch den Benutzer oder das Umleiten von Aufträgen durch die Systembetreuung wird dadurch nicht berührt.

### **RDIR=\*STD / \*YES**

Ein Geräteverwalter darf den RSO-Auftrag auf einen anderen Drucker umleiten.

### **RDIR=\*NO**

Ein Geräteverwalter darf den RSO-Auftrag nicht auf einen anderen Drucker umleiten.

**RDIR=<var: enum-of \_redir\_type\_set:1>**

Die Vereinbarung bzgl. des Umleitens des RSO-Auftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und gewünschter Funktion:

0	*STD
1	*YES
2	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 181</a> .	

**TOPOFF=\*IGNORE / <integer -255..255>**

Legt den Abstand der ersten zu druckenden Zeile vom oberen Rand der physikalischen Seite in Millimetern fest.

Dieser Operand ist nur für die Drucker 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9021, 9022-200, 9025, 9026-PCL und 9026-RENO zulässig.

**TOPOFF=\*IGNORE**

Die Druckersteuerung setzt keinen Abstand zwischen der ersten Zeile und dem oberen Seitenrand fest. Die Druckervoreinstellung oder die Einstellung durch die Prologdatei legen den Abstand fest, der beim Ausdrucken des Dokuments wirksam wird.

**TOPOFF=<integer -255..255>**

Positive Werte verschieben die erste zu druckende Zeile nach unten. Negative Werte verschieben die erste zu druckende Zeile nach oben. Der zulässige Wertebereich ist -255 bis +255.

**LEFTOFF=\*IGNORE / <integer -255..255>**

Legt den Abstand der ersten zu druckenden Zeile vom linken Rand der physikalischen Seite in Millimetern fest.

Dieser Operand ist nur für die Drucker 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9021, 9022-200, 9025, 9026-PCL und 9026-RENO zulässig.

**LEFTOFF=\*IGNORE**

Die Druckersteuerung setzt keinen Abstand zwischen der ersten Zeile und dem linken Seitenrand fest. Die Druckervoreinstellung oder die Einstellung durch die Prologdatei legen den Abstand fest, der beim Ausdrucken des Dokuments wirksam wird.

**LEFTOFF=<integer -255..255>**

Positive Werte verschieben die erste zu druckende Zeile nach rechts. Negative Werte verschieben die erste zu druckende Zeile nach links. Der zulässige Wertebereich ist -255 bis +255.

**URL=(*urlptr,urlen,prtype*)**

Bezeichnet den IPP-Drucker, der durch seine URL adressiert wird.

***urlptr*: <var: pointer>**

Bereich, der die Drucker-URL enthält

***urlptr*: \*NONE**

Standard-Einstellung. Es wurde keine Drucker-URL angegeben.

***urlen*: <var: int 0..65535>**

Länge des Bereichs.

***urlen*: <integer 1..1023>**

Länge des Bereichs.

***urlen*: \*NONE**

Standard-Einstellung. Es wurde keine URL angegeben.

***prtype*: <var: enum-of:1 \_PRTYPE\_SET>**

Der Druckertyp wird nicht direkt durch einen Operandenwert angegeben. Stattdessen wird er indirekt durch ein Feld mit konstantem Inhalt (Equate) angegeben. Es kann ein Integer-Wert in der Konstanten oder im zugehörigen Feld gespeichert werden. Folgende Beziehung besteht zwischen den Werten und den gewünschten Funktionen:

Wert	Druckertyp
72	*4850_PCL
73	*4830_PCL
74	*4818_PCL
75	*9000_EPSQ
76	*9000_EPLQ
77	*9000_EPFX
78	*9000_PRO
79	*9000_PS
7B	*9046
7D	*9045_ANSI
7E	*9015
7F	*4825_PCL
80	*4822_PCL
84	*9001
88	*9025
8A	*9013
8B	*9000

Wert	Druckertyp
8C	*9001_31
8D	*9011
8E	*9012
8F	*9022
A0	*9014
A1	*9021
A2	*9022_200
A4	*9000_PCL
A6	*4824_PCL
A7	*9026_RENO
A8	*9026_PCL
A9	*4812
AA	*9097
AB	*4011
AC	*4813
AD	*4821_PCL
AE	*2030_PCL

*prtype:* \*4850\_PCL | \*4830\_PCL | \*4818\_PCL | \*9000\_EPSQ | \*9000\_EPLQ | \*9000\_EPFX | \*9000\_PRO | \*9000\_PS | \*9046 | \*9045\_ANSI | \*9015 | \*4825\_PCL | \*4822\_PCL | \*9001 | \*9025 | \*9013 | \*9000 | \*9001\_31 | \*9011 | \*9012 | \*9022 | \*9014 | \*9021 | \*9022\_200 | \*9000\_PCL | \*4824\_PCL | \*9026\_RENO | \*9026\_PCL | \*4812 | \*9097 | \*4011 | \*4813 | \*4821\_PCL | \*2030\_PCL

RSO-Gerätetyp des Zieldruckers.

*prtype:* **\*NONE**

Standard-Einstellung. Es wurde keine URL angegeben.

**Beschreibung der Parameterliste**

\* parameter list description

SROAMDLS DS 0F

SROAHDR FHDR MF=(C,SROA),EQUATES=NO Standard header

SROACHRAS DS FL1 Specifies whether all or only  
\* specific character set  
\* attributes are to be used  
\* for the print-job. These  
\* attributes include character  
\* type, near letter quality,  
\* color, etc. This operand  
\* does not apply to header and  
\* trailer pages.

\* \_char\_att\_type\_set

SROACSTD EQU 0

SROACALL EQU 1

SROACRST EQU 2

\*

SROARDIR DS FL1 Specifies whether a device  
\* administrator may redirect  
\* print-job to another  
\* printer. Redirection of  
\* print-job by users or  
\* system-administration is not  
\* affected by this operand.

\* \_redir\_type\_set

SROARSTD EQU 0

SROARYES EQU 1

SROARNO EQU 2

\*

SROASPEC DS AL1 specified; for compatibility  
\* with MODJRSO

SROACSPC EQU X'80'

SROARSPC EQU X'40'

SROATSPC EQU X'20'

SROALSPC EQU X'10'

SROAUNUS EQU X'0F'

SROAFIL1 DS CL1 slack byte

SROATOPD DS H Specifies the length of the  
\* white band that must be  
\* inserted

\*

SROALEFO DS H Specifies the width of the  
\* white band that must be  
\* inserted

\*

SROAUNS1 DS XL8 unused

SROA# EQU \*-SROAHDR

## PRNTDPC - Dprint-Parameter für PRNTDOC-Makro definieren

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprachen:** Assembler, C

**Makrotyp:** S

Mit diesem Makro werden die Dprint-Parameter für den PRNTDOC-Makro definiert.

### Format

Operation	Operanden
PRNTDPC	VARIANT= <u>001</u> / <c-string 3..3> ,FORMAT= <u>*STD</u> / <var: char: 63> / <c-string 1..63> ,CLUSTER= <u>*LOCAL-CLUSTER</u> / <var: char: 8> / <c-string 1..8> .RESLOC= <u>*STD</u> / *HOME / *SERVER / <var: enum-of_resloc_set:1> ,OUTFORM= <u>*STD</u> / <c-string 1..63>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**FORMAT=\*STD / <var: char: 63> / <c-string 1..63>**

Art des Dokument-Inhalts.

**\*STD**

Standardmäßig wird keine spezielle Formatbezeichnung angenommen.

**<var: char: 63>**

Name des Feldes mit der Formatbezeichnung.

**<c-string 1..63>**

Formatbezeichnung; Direktangabe.

**CLUSTER=\*LOCAL-CLUSTER / <var: char: 8> / <c-string 1..8>**

Angabe des Dprint-Clusters.

**\*LOCAL-CLUSTER**

Standardmäßig wird der lokale Cluster angenommen.

**<var: char: 8>**

Name des Feldes mit dem Cluster-Namen.

**<c-string 1..8>**

Cluster-Name; Direktangabe.

**RESLOC=**

Lage der Ressourcen, die für den Druck verwendet werden.

**\*STD**

Standardmäßig werden die Werte aus der SPOOL-Parameterdatei angenommen.

**\*HOME**

Die Client-Ressourcen werden verwendet.

**\*SERVER**

Die Server-Ressourcen werden verwendet.

**<var: enum-of\_resloc\_set:1>**

Name des Equates für den Operandenwert. Nur in Verbindung mit MF=M möglich.

**OUTFORM=**

Nur relevant für die Interoperabilität mit SPOOL in UNIX-Systemen. Gibt an, welche Druckerspache für den Druckauftrag benutzt werden soll.

**\*NONE**

Es wurde kein Ausgabeformat angegeben. Das Dokument wird ohne Änderungen an das UNIX-System übermittelt. Es wird angenommen, dass der Drucker im UNIX-System den Formatnamen kennt, der in der Operandenstruktur DOCFORM=\*SPECIAL\_FORMAT(...) angegeben wurde.

**<c-string: 1..63 with-low>**

Formatname, den der im Operanden PRNAME der Makros PRNTDOC und MODPJAT angegebenen Drucker im UNIX-System kennt und unterstützt. Siehe Operandenbeschreibung auf [Seite 94](#) und [Abschnitt „MODPJAT - Attribute eines Druckauftrags modifizieren“ auf Seite 220](#).

**Beschreibung der Parameterliste**

```

*   parameter list description
SDCAPL   DS    0F
SDCAHDR  FHDR  MF=(C,SDCA),EQUATES=NO   Standard header
SDCACLUS DS    CL8                       cluster name to which the
*                                           print-job is submitted
SDCAFMT  DS    CL63                       document content type
SDCARESL DS    FL1                       location of the resources
*                                           used to process the
*                                           print-job is submitted
*   _resloc_set
SDCARSTD EQU    0
SDCARHOM EQU    1
SDCARSRV EQU    2
*
SDCASP1  DS    AL1                       specified
SDCASP11 EQU   X'80'
SDCASP12 EQU   X'40'
SDCASP13 EQU   X'20'
SDCASP14 EQU   X'10'
SDCASP1F EQU   X'0F'
SDCAOUTF DS    CL63                       printer specific language
SDCAFIL1 DS    CL4                       slack bytes
SDCA#    EQU   *-SDCAHDR

```



**C-Headerfile**

```

/* _resloc_set */
/* ENUM _resloc_set */
#define PRNTDPCresloc_std 0 /* */
#define PRNTDPCresloc_home 1 /* */
#define PRNTDPCresloc_serv 2 /* */

/* parameter list description */
struct PRNTDPC_pl_md1 {

    /* Standard header */
    struct ESMFHDR hdr;
    char cluster[8]; /* cluster name to which the */
                    /* print-job is submitted */
    char format[63]; /* document content type */
    unsigned char resloc; /* location of the resources */
                        /* used to process the */
                        /* print-job is submitted */

    /* specified */
    struct /* specified */ {
        char spec_format: 1; /* */
        char spec_cluster: 1; /* */
        char spec_resloc: 1; /* */
        char spec_outfrm: 1; /* */
        char spec_free: 4; /* */
    } specified;
    char out_format[63]; /* printer specific language */
    char filler1[4]; /* slack bytes */
};

```

## Struktur-Layout

Offset	Identifikation	Datentyp:Wert	Bedeutung/Bemerkung
FW	PRNTDPC_pl_mdl	STRUCT:84	Beschreibung d. Parameterliste
000	hdr	ESMFHDR:8	Standard-Header
008	cluster	CHAR:8	Cluster-Name, an den der Druckauftrag abgegeben wird
010	format	CHAR:63	Inhaltstyp des Dokuments
04F	resloc	ENUM:1	Lage der für den übertragenen Druckauftrag verwendeten Ressourcen
	std	0	Assembler-Liste: ASS:SDCARESLOC_STD
	home	1	Assembler-Liste: ASS:SDCARESLOC_HOME
	server	2	Assembler-Liste: ASS:SDCARESLOC_SERV
050	specified	FLAG:1	spezifiziert
00	format	BIT:1	Assembler-Liste: ASS:SDCASPEC_FORMAT
01	cluster C-F:spec_cluster	BIT:1	Assembler-Liste: ASS:SDCASPEC_CLUSTER
02	resloc C-F:spec_cluster	BIT:1	Assembler-Liste: ASS:SDCASPEC_RESLOC
03	spec_free	BIT:5	
051	filler1	CHAR:3	Slack-Bytes

## PRNTSPS - SPS-APA-Parameter für PRNTDOC-Makro definieren

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Mit diesem Makro werden für SPS-APA spezifische Operanden gesetzt.

### Format

Operation	Operanden
PRNTSPS	VARIANT= <u>001</u> / <c-string3..3>
	FONTS= <u>*STD</u> / array(4): <var: char: 8: alphanum_name_1..8>
	PAGEDEF= <u>*STD</u> / <var: char: 8: alphanum_name_1..8>
	FORMDEF= <u>*STD</u> / <var: char: 8: alphanum_name_1..8>
	TRC= <u>*STD</u> / <var: enum-of _trc_set:1> / *YES / *NO
	MSGPAGE= <u>*STD</u> / <var: enum-of _msgpage_set:1> / *YES / *NO
	FRONTOVL= <u>*STD</u> / <var: char: 8:alphanum_name_1..8>
	BACKOVL= <u>*STD</u> / <var: char: 8:alphanum_name_1..8>

## Operandenbeschreibung

### **VARIANT=<c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **FONTS=\*STD / array(4): <var: char: 8: alphanum\_name\_1..8>**

Nur für lokalen Spool.

Vereinbart die Fonts, die bei Verwendung des Operanden TABLE-REFERENCE-CHAR von SPS auszuwerten sind.

### **FONTS=\*STD**

Aus dem gewünschten Formular wird der Standardfont für diesen Druckertyp gewählt. Er kann mit SHOW-SPOOL-FORMS abgefragt werden.

### **FONTS=array(4): <var: char: 8: alphanum\_name\_1..8>**

Name der Fonts, die mit den TRC-Werten in der Druckdatei zu verknüpfen sind. Jeder TRC-Wert repräsentiert jeweils einen bestimmten Font.

Unabhängig von der Anzahl unterschiedlicher TRC-Werte in der Datei können maximal vier Fonts, die in Form einer Liste anzugeben sind, für den Ausdruck der Datensätze verwendet werden. Höhere TRC-Werte als X'03' (entsprache dem vierten Element der Liste) verweisen dabei automatisch auf den in der Liste erstgenannten Font.

### **PAGEDEF=\*STD / <var: char: 8: alphanum\_name\_1..8>**

Nur für lokalen SPOOL.

Gibt an, welche Seiten-Definition für den Ausdruck auf LP65- oder APA-Druckern benutzt werden soll.

### **PAGEDEF=\*STD**

Für APA-Drucker:

Die Page Definition, die dem in FORM-NAME angegebenen Formular zugeordnet ist, wird benutzt.

Für LP65-Drucker:

Nummer der PCL-Datei, mit der die Druckdatei ausgegeben werden soll. Nur die Druckdatei selbst wird mit dieser PCL-Datei ausgedruckt. Deck- und Schlussblatt werden über die PCL-Datei gesteuert, die im Formular der SPOOL-Parameterdatei definiert ist.

### *Hinweise*

- Eine PCL-Datei, die es unmöglich macht, eine andere PCL-Datei über ein Kanalkommando auszuwählen, darf nicht verwendet werden.
- Nach dem Starten der angegebenen PCL-Datei wird der Betriebsmodus des Druckers überprüft. Ist der EXCCW-Modus aktiv und tritt während des Druckvorganges ein Fehler auf, ist der Wiederaufsetzpunkt der letzte SECTION-Satz; wurde keine Angabe bei dem Operanden SECTION gemacht, wird die Datei von Anfang an erneut ausgedruckt.

**PAGEDEF=<var: char: 8: alphanum\_name\_1..8>**

Nur für APA-Drucker:

Die Seiten-Definition mit dem angegebenen Namen soll benutzt werden. Diese muss sich in der SPSSLIB befinden.

*Hinweis*

Die ersten beiden Zeichen des angegebenen Namens müssen „P1“ sein. Andernfalls wird das Makro abgewiesen.

**FORMDEF=\*STD / <var: char: 8: alphanum\_name\_1..8>**

Nur für lokalen SPOOL.

Gibt an, welche Format-Definition für den Ausdruck auf APA-Druckern benutzt werden soll.

**FORMDEF=\*STD**

Die Format-Definition, die dem in FORM-NAME angegebenen Formular zugeordnet ist, wird benutzt.

**FORMDEF=<var: char: 8: alphanum\_name\_1..8>**

Die Format-Definition mit dem angegebenen Namen soll benutzt werden. Diese muss sich in der SPSSLIB befinden.

*Hinweis*

Die ersten beiden Zeichen des angegebenen Namens müssen „F1“ sein. Andernfalls wird das Makro abgewiesen.

**TRC=\*STD / <var: enum-of \_trc\_set:1> / \*YES / \*NO**

Legt fest, ob der Anwender die Auswahl von Fonts zur Druckseitengestaltung über Steuerzeichen im Text vornimmt. Diese Auswahl kann mit Hilfe der TRC-Steuerzeichen (Table Reference-Character) vorgenommen werden, die innerhalb der Datensätze auf eine Liste von Fonts verweisen.

Die Liste kann entweder Bestandteil der Seitendefinition PAGE-DEFINITION sein oder mit Hilfe des Operanden CHARACTER-SETS bekannt gegeben werden. Die ausführliche Erläuterung zur Anwendung der TRC ist im Handbuch „[SPOOL \(BS2000/OSD\)](#)“ zu finden.

**TRC=\*STD****TRC=\*YES**

Innerhalb der Druckdatei befinden sich Font-Auswahl-Steuerzeichen (TRC), die von SPS auszuwerten sind. Ein TRC-Wert repräsentiert jeweils einen bestimmten Font, mit dem der betreffende Datensatz auszudrucken ist. Die einzelnen Werte (von X'00' bis X'0F') sind entsprechend ihrer Reihenfolge mit den Einträgen in der FONTLIST (Fontliste der Seitendefinition PAGEDEF) zu verknüpfen oder sind entsprechend ihrer Reihenfolge explizit mit im Operanden CHARACTER-SETS angegebenen Fonts zu verknüpfen. Unabhängig von der Anzahl unterschiedlicher TRC-Werte in der Datei können maximal vier Fonts, die in Form einer Liste anzugeben sind, für den Ausdruck der Datensätze verwendet werden.

Höhere TRC-Werte als X'03' (entspräche dem vierten Element der Liste) verweisen dabei automatisch auf den in der Liste erstgenannten Font.

**TRC=\*NO**

Innerhalb der Druckdatei sind keine Font-Auswahl-Steuerzeichen (TRC) hinterlegt oder diese sollen nicht ausgewertet werden.

**MSGPAGE=\*STD / <var: enum-of \_msgpage\_set:1> / \*YES / \*NO**

Gibt an, ob die APA-Meldungsseite gedruckt werden soll oder nicht. Die APA-Meldungsseite enthält Fehlermeldungen und Warnungen. Falls kein APA-Drucker benutzt wird, wird die Angabe ignoriert.

**FRONTOVL=\*STD / <var: char: 8:alphanum\_name\_1..8>**

Vereinbart, ob ein APA-Overlay für die Vorderseite verwendet werden soll.

**FRONTOVL=\*STD**

Es werden keine APA-Overlays für die Ausgabe benutzt, außer wenn in der zugehörigen Form Definition ein APA-Overlay angegeben ist.

**FRONTOVL=<var: char: 8:alphanum\_name\_1..8>**

Name des APA-Overlays, das für die Vorderseite verwendet werden soll.

**BACKOVL=\*STD / <var: char: 8:alphanum\_name\_1..8>**

Vereinbart, ob ein APA-Overlay für die Rückseite verwendet werden soll.

**BACKOVL=\*STD**

Es werden keine APA-Overlays für die Ausgabe benutzt, außer wenn in der zugehörigen Form Definition ein APA-Overlay angegeben ist.

**BACKOVL=<var: char: 8:alphanum\_name\_1..8>**

Name des APA-Overlays, das für die Rückseite verwendet werden soll.

**Beschreibung der Parameterliste**

* parameter list description			
SPSAPL	DS	0F	
SPSAHDR	FHDR	MF=(C,SPSA),EQUATES=NO	Standard header
SPSAFNT	DS	4CL8	When TRCs are used, allows to reference a list of fonts
*			
SPSAFNT#	EQU	4	
SPSAPAGD	DS	CL8	Element name of the page definition to be used.
*			
SPSAFRMD	DS	CL8	Element name of the format definition to be used.
*			
SPSATRC	DS	FL1	Specifies whether the file to print contains TRCs.
*			
* _trc_set			
SPSATRCY	EQU	1	
SPSATRCN	EQU	2	
SPSATRCS	EQU	4	
*			
SPSAMSGP	DS	FL1	
* _msgpage_set			
SPSAMSGY	EQU	1	
SPSAMSGN	EQU	2	
SPSAMSGS	EQU	4	
*			
SPSASP1	DS	AL1	specified
SPSASP11	EQU	X'80'	
SPSASP12	EQU	X'40'	
SPSASP13	EQU	X'20'	
SPSASP14	EQU	X'10'	
SPSASP15	EQU	X'08'	
SPSASP1F	EQU	X'07'	
SPSAFIL1	DS	CL1	slack byte
SPSA#	EQU	*-SPSAHDR	

## SNPPRNT - Notification-Parameter für PRNTDOC definieren

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprachen:** Assembler, C, CPP

**Makrotyp:** S

Das Makro SNPPRNT beschreibt die Parameterliste für die Unterstützung des Notification Service.

### Format

Operation	Operanden
SNPPRNT	MF = D / I / M ,PREFIX = SNPPRNT / <name> ,MACID = <name 1..3> ,PARAM = <name 1..27> ,NOTIFICATION = <u>*STD</u> / *NO / *YES ,MTH_NAME = *MAIL / <c-string 1..8> / <var: char 1..8> ,OBJ_ATTR_TYP = <u>*NONE</u> / *ALL / *LIST ,OBJ_ATTR = array(20): elem: <c-string 1..64> / <var:char 1..64> ,EVT_NAME_TYP = <u>*ALL</u> / *LIST ,EVT_NAME = array(20): elem: <c-string 1..24> / <var:char 1..24> ,USER_DATA = <u>*NONE</u> / <c-string 1..63> / <var: char 1..63> ,RECIPIENT_ADDR = <c-string 1..224> / <var: char 1..224>

### Beschreibung der Operanden

#### MF=D / I / M

Typ des Makroaufrufs. Weitere Informationen finden Sie im Handbuch „[Makroaufrufe an den Ablaufteil](#)“. Mögliche Werte für die verschiedenen Programmiersprachen:

Assembler:	C, D, L, M
C:	--
CPP:	D, I, M

#### *Hinweis*

Sie können mit MF=M nur Pointer-Variablen angeben ( „var:pointer“ Operandenwert). Mit MF=D muss zusätzlich eine Liste erzeugt werden.



**PREFIX=SNPPRNT / <name>**

Legt das erste Zeichen der Feldnamen und Equates fest.

Voreingestellte Werte für die verschiedenen Programmiersprachen:

Assembler:	S
C:	--
CPP:	SNPPRNT

**MACID=<name 1..3>**

Legt das erste bis vierte Zeichen (inklusive) der Feldnamen und Equates fest.

Voreingestellte Werte für die verschiedenen Programmiersprachen:

Assembler:	PRT
C:	--
CPP:	--

**PARAM=<name 1..27>**

Legt die Adresse der Operandenliste fest (nur erlaubt bei den MF-Formaten 2 und 3). Weitere Informationen finden Sie im Handbuch „[Makroaufrufe an den Ablaufteil](#)“.

**NOTIFICATION=\*STD / \*NO / \*YES**

Wählt die Notification-Verarbeitung für die zugeordneten Druckaufträge aus.

**NOTIFICATION=\*STD**

Für den aktuellen Druckauftrag werden Benachrichtigungen verschickt, wenn der Besitzer dieses Druckauftrags vorher permanente Subskriptionen in die Notification-Ressourcendatei eingetragen hat.

**NOTIFICATION=\*NO**

Die Notification-Verarbeitung wird für diesen Druckauftrag ausgeschaltet. Es werden keine Benachrichtigungen erzeugt, auch wenn in der Notification-Ressourcendatei gültige Subskriptionen eingetragen sind, die dem Besitzer dieses Druckauftrags gehören. Die Benachrichtigungen, die für Subskriptionen anderer privilegierter Benutzer erzeugt werden, werden jedoch nicht ausgeschaltet.

**NOTIFICATION=\*YES**

Dieser Operand erzeugt eine temporäre Subskription-Ressource. Diese Subskription bleibt bestehen, solange der aktuelle Druckauftrag existiert. Dieser Operand erlaubt es dem Benutzer, einem bestimmten Druckauftrag eine Subskription-Ressource zuzuordnen.

**MTH\_NAME=\*MAIL / <c-string 1..8> / <var: char 1..8>**

Gibt die Notification-Zustellmethode an.

**OBJ\_ATTR\_TYP=\*NONE / \*ALL / \*LIST**

Legt die Objekt-Attribute fest, die den Benachrichtigungen zugeordnet sind. Objekten, z.B. Druckaufträgen, für die Benachrichtigungen gesendet werden, können Attribute zugeordnet sein. Der Benutzer möchte vielleicht, dass eines oder mehrere dieser zugeordneten Attribute mit einer bestimmten Benachrichtigung zurückgeliefert werden. Im Allgemeinen können Benachrichtigungen jedes beliebige Attribut enthalten, das dem Objekt zugeordnet ist, das die Benachrichtigung aussendet.

**OBJ\_ATTR\_TYP=\*NONE**

Es wird kein Attribut ausgewählt.

**OBJ\_ATTR\_TYP=\*ALL**

Alle Attribute, die dem Druckauftrag zugeordnet sind, werden ausgewählt.

**OBJ\_ATTR\_TYP=\*LIST**

Einige Attribute werden ausgewählt. Der Operand OBJ\_ATTR muss angegeben werden.

**OBJ\_ATTR=array(20): elem: <c-string 1..64> / <var:char 1..64>**

Gibt eine explizite Liste von Attributen an. Nur relevant mit OBJ\_ATTR\_TYP=\*LIST.

**EVT\_NAME\_TYP=\*ALL / \*LIST**

Wählt den Typ der Ereignisse aus, die dem Druckauftrag zugeordnet sind.

**EVT\_NAME\_TYP=\*ALL**

Alle Ereignisse, die dem Druckauftrag zugeordnet sind, werden ausgewählt.

**EVT\_NAME\_TYP=\*LIST**

Einige Ereignisse, die dem Druckauftrag zugeordnet sind, werden ausgewählt. Der Operand EVT\_NAME muss angegeben werden.

**EVT\_NAME=array(20): elem: <c-string 1..24> / <var:char 1..24>**

Gibt eine explizite Liste von subskribierten Ereignissen an. Nur relevant mit EVT\_NAME\_TYP = \*LIST.

**USER\_DATA=\*NONE / <c-string 1..63> / <var: char 1..63>**

Enthält opake Daten, die einige Zustellmethoden in die Benachrichtigungen einschließen.

**USER\_DATA=\*NONE**

Es sind keine Benutzerdaten angegeben.

**USER\_DATA=<c-string 1..63>**

Gibt den Wert der Benutzerdaten an.

**USER\_DATA=<var: char 1..63>**

Gibt den Wert der Benutzerdaten an.

**RECIPIENT\_ADDR=<c-string 1..224> / <var: char 1..224>**

Zustelladresse für die Benachrichtigungen je nach ausgewählter Methode.

## Struktur-Layouts

### SNPPRNT (Assembler)

Für Assembler hat der Makro SNPPRNT das folgende Struktur-Layout:

```
*****
* BEGIN-INTERFACE    SNPPRNT
*
* TITLE              (/ Notification PRINT-DOC description /)
* NAME               SNPPRNT
* DOMAIN             SPOOL
* LANGUAGE           ASS
* COPYRIGHT          (C) Fujitsu Siemens Computers GmbH 2003
*                   ALL RIGHTS RESERVED
* COMPILATION-SCOPE RESTRICTED
* INTERFACE-TYPE     CALL
* RUN-CONTEXT        TU
*
* PURPOSE            (/ Layout of notification part of PRINT-DOC api /)
*
* SYNTAX              (/ Syntax Variant 1:
*                   SNPPRNT MF = C|D|M|L
*                   , PREFIX      = [S] | <name>
*                   , MACID       = [PRT] | <name>
*                   , EQUATES     = [YES] | NO
*                   , NOTIFICATION= *STD |
*                   *NO |
*                   *YES |
*                   default *STD
*                   , MTH_NAME    = *MAIL |
*                   <c-string 1..8> |
*                   <var: char 1..8>
*                   , OBJ_ATTR_TYP= *NONE |
*                   *ALL |
*                   *LIST |
*                   default *NONE
*                   , OBJ_ATTR   = array(20)
*                   elem: <c-string 1..64> |
*                   <var: char 1..64>
*                   , EVT_NAME_TYP= *ALL |
*                   *LIST |
*                   default *ALL
*                   , EVT_NAME   = array(20)
*                   elem: <c-string 1..24> |
*                   <var: char 1..24>
*                   , USER_DATA = *NONE |
*                   <c-string 1..63> |
```

```
*          <var: char 1..63> |
*          default *NONE
*          , RECIPIENT_ADDR= <c-string 1..224> |
*          <var: char 1..224> /)
*
* REMARKS          (/ Welcome To Notification Service /)
*
* END-INTERFACE    SNPPRNT.
*****
```

### **SNPPRNT.H (C)**

Siehe Assembler-Beschreibung

### **SNPPRNTC (CPP)**

Siehe Assembler-Beschreibung

## SSVBARC - Datenkonvertierung für Bar Codes

**Benutzergruppe:** Privilegierter Benutzer

**Programmiersprachen:** Assembler

**Makrotyp:** M

Dieses Makro stellt einen Standardweg für den Aufruf des Bar Code-Programms zur Verfügung. Das Bar Code-Programm (LLM) konvertiert ein Zeichen in die entsprechende Darstellung als Bar Code.

### Format:

Operation	Operanden
SSVBARC	MF = C / D / L / M ,PREFIX= <u>S</u> / <name 1..1> ,MACID= <u>BRC</u> / <name 1..3> ,EQUATES= <u>YES</u> / NO ,INDTAADR=<var: pointer> ,INDTALEN=<var: int 0..2147483647> ,OUTDTAADR=<var: pointer> ,OUTDTALEN=<var: int 0..2147483647>

### Beschreibung der Operanden

**MF=C / D / L / M**

Typ des Makroaufrufs. Weitere Informationen finden Sie im Handbuch „[Makroaufrufe an den Ablaufteil](#)“.

**PREFIX=S / <name 1..1>**

Legt das erste Zeichen der Feldnamen und Equates fest.

**MACID=BRC / <name 1..3>**

Legt das zweite bis vierte Zeichen (inklusive) der Feldnamen und Equates fest.

**EQUATES=YES / NO**

Legt fest, ob Equates generiert werden oder nicht.

**INDTAADR=<var: pointer>**

Gibt die Adresse der Zeichenfolge an, die konvertiert werden soll.

**INDTALEN=<var: int 0..2147483647>**

Gibt die Anzahl der Zeichen an, die konvertiert werden sollen.

**OUTDTAADR=<var: pointer>**

Gibt die Adresse des Bereichs an, in den der Bar Code (die konvertierte Zeichenfolge) geschrieben werden soll.

**OUTDTALEN=<var: int 0..2147483647>**

Gibt die Länge des Bereichs für den Bar Code an.

**Returncodes**

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	Verarbeitung fehlerfrei beendet
00	01	0001	Fehler bei der Speicheranforderung
00	02	0001	Puffer zu klein

**Strukturbeschreibung**

SSVBARC (ASS)

\*\*\*\*\*

```
* BEGIN-INTERFACE   SSVBARC
*
* TITLE              (/ TRANSLATE DATA FOR BARCODE PARAM LIST /)
* NAME               SSVBARC
* DOMAIN             SPOOL
* LANGUAGE           ASS
* COPYRIGHT          (C) Fujitsu Siemens Computers GmbH 2004
*                   ALL RIGHTS RESERVED
* COMPILATION-SCOPE RESTRICTED
* INTERFACE-TYPE    CALL
* RUN-CONTEXT       TPR
*
* PURPOSE           (/ TRANSLATE DATA FOR BARCODE PARAM LIST /)
*
* SYNTAX            (/ Syntax Variant 1:
*                   SSVBARC MF = C|D|L|M
*                   , PREFIX   = [S] | <name>
*                   , MACID    = [BRC] | <name>
*                   , EQUATES  = [YES] | NO
*                   , INDTAADR = <var: pointer>
*                   , INDTALEN = <var: int 0..2147483647>
*                   , OUTDTAADR = <var: pointer>
*                   , OUTDTALEN = <var: int 0..2147483647> /)
*
* END-INTERFACE     SSVBARC.
*****
```

---

## 3 Makros für die Ausgabe von Informationen

In diesem Kapitel ist das Makro SPSINF beschrieben. Mit diesem Makro können Sie sich über den Inhalt der SPOOL-Parameterdatei informieren.

### SPSINF - Informationen anfordern

**Benutzergruppe:** Systembetreuung, nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** O

Der Makro SPSINF informiert über den Inhalt der SPOOL-Parameterdatei. Folgende Informationen werden ausgegeben:

- Anzahl der Einträge eines Typs (Geräte-, Formular-, Zeichensatz-Einträge); siehe auch Aufbau der SPOOL-Parameterdatei (siehe Handbuch „[SPSERVE \(BS2000/OSD\)](#)“).
- Inhalt des Eintrags eines angegebenen Typs.
- SPOOL-globale Voreinstellungen.
- Liste der Gerätenamen, Namen der Zeichensätze, Formulare oder Gerätepools.
- Inhalt aller Einträge für alle Typen.

Die verlangte Information wird über eine Kurzbezeichnung identifiziert und in einen Ausgabebereich übertragen. Die Liste auf [Seite 205](#) enthält die anzugebende Kurzbezeichnung, die dafür erforderliche Mindestlänge des Ausgabebereichs und den Namen des Makros, der eine Beschreibung des jeweiligen Ausgabebereichs generiert.

Die Größe des Ausgabebereichs ist generell davon abhängig, ob nur Informationen über einen Eintrag aus der SPOOL-Parameterdatei gewünscht werden, oder ob eine Liste über mehrere oder alle Einträge ausgegeben werden soll. In einigen Fällen wird der benötigte Speicherplatz durch die Makrofunktion angefordert (interner REQM); die Freigabe erfolgt aber durch den Benutzer. Die Mindestlänge des Ausgabebereichs kann der Tabelle auf [Seite 205](#) oder den DSECTs für die Ausgabebereiche entnommen werden. Ist der Ausgabebereich zu klein für die Menge der angeforderten Informationen, kann mit einem oder mehreren weiteren Makro-Aufrufen (mit der gleichen Parameterliste) die Restmenge abgefragt werden.

Die Makros SPSxxxD und der Makro SPSPL generieren Beschreibungen der Ausgabebereiche; die Makros SPSxxxD als Dsect oder Datenbereich, der Makro SPSPL nur als Datenbereich. Der vollständige Name für SPSxxxD, sowie mögliche Operandenwerte für SPSPL PL=... sind der Tabelle im Abschnitt „[Kurzbezeichnung für Operanden INFO, Mindestlängen und Erläuterungen](#)“ auf [Seite 205](#) zu entnehmen.

Wird ein SPSINF-Aufruf abgesetzt, während SPOOL nicht geladen ist, wird im Register 15 der Returncode x'01000000' zurückgegeben (Dies trifft für alle SPOOL-Aufrufe zu.)



**Format**

Operation	Operanden
SPSINF	,INFO=info  [,ITEM= $\left. \begin{matrix} \text{name} \\ \text{(adr)} \end{matrix} \right\}$ ] ]  [,DVCTYP=gerät]  [,SEQN=zahl]  [, $\left\{ \begin{matrix} [\text{OUTPL=adr}][\text{OUTL=länge}] \\ [\text{BUFFER=adr}][\text{BUFL=länge}] \end{matrix} \right\}$ ] ]  [,MF= $\left. \begin{matrix} \text{E} \\ \text{(E,adr)} \\ \text{L} \\ \text{(L,p)} \\ \text{C} \\ \text{(C,p)} \\ \text{D} \\ \text{(D,p)} \end{matrix} \right\}$ ] ]
SPSxxxD	MF=(L[,p]) / (C[,p]) / (D[,p])
SPSPL	PL=liste,P=p

**Operandenbeschreibung**

- INFO                    beschreibt über eine Kurzbezeichnung, welche Informationen ausgegeben werden sollen. Die Kurzbezeichnung kann als Name oder Zahl (in Klammern) angegeben werden.
- =info                    info = Kurzbezeichnung; siehe Liste am Ende der Operandenbeschreibung. Voreinstellung: info=SPSGEN.

*Hinweis*

Ist INFO=SPSDVC angegeben und keine Angabe bei dem Operanden ITEM gemacht, wird trotzdem ein Geräteeintrag ausgegeben: der generelle Standard-Geräteeintrag (DVCTYP=X'00').

ITEM bezeichnet den auszugebenden Eintrag durch Angabe eines Namens oder der Adresse eines Feldes mit dem Namen. Der Name kann Wildcards enthalten (Beschreibung der Wildcards siehe [Abschnitt „Musterzeichen“ auf Seite 608](#)). Die Länge des Namens ist begrenzt; sehr lange Namen können mit Hilfe von Wildcards und Eintrag in das Feld angegeben werden.

Typ	Länge des Namens	
	ohne Wildcards	mit Wildcards
Gerät	≤ 8 Zeichen	≤ 8 Zeichen bei direkter Angabe bzw.
Gerätepool	≤ 8 Zeichen	≤ 24 Zeichen bei Angabe im Feld
Zeichensatz	≤ 3 Zeichen	
Formular	≤ 6 Zeichen	

Die Verwendung von Wildcards ist nicht erlaubt, wenn bei INFO=... die Kurzbezeichnungen SPSPOL, SPSFRM, SPSDVC oder SPSPOL angegeben werden.

=name Name eines Geräts, Gerätepools, Zeichensatzes oder Formulars.

=(adr) adr = Adresse des Feldes mit dem Namen. Feldlänge ≥ 24 Bytes; Eintrag linksbündig, mit Leerzeichen (X'40') auf 24 Bytes auffüllen.

DVCTYP bezeichnet einen Gerätetyp. DVCTYP muss spezifiziert werden, wenn bei INFO=SPSDVC angegeben wird und bei ITEM=... keine Angabe erfolgte. Es werden die Formular-Einträge für den angegebenen Gerätetyp ausgegeben. In Verbindung mit INFO=SPSFRM gilt folgende Voreinstellung: Der erste Formular-Eintrag in der SPOOL-Parameterdatei wird ausgegeben.

=typ typ = Bezeichnung des Gerätetyps.

SEQN muss in Verbindung mit INFO=SPSPOL oder INFO=SPSDVP angegeben werden und bezeichnet den auszugebenden Geräteeintrag innerhalb des Gerätepools oder aus der Menge der Geräteeinträge (INFO=SPSDVP).

=zahl bezeichnet die Stelle des Eintrags innerhalb des Gerätepools, oder seine Stellung bei den Geräteeinträgen. (1. Eintrag, 2. Eintrag, 3. Eintrag, ...).  
 $0 \leq \text{zahl} \leq 65535$ ; bei zahl=0 wird die Anzahl der Geräte für jeden Gerätetyp ausgegeben. (zahl=0 ist nur mit INFO=SPSPOL sinnvoll).

OUTPL	bezeichnet die Adresse des Ausgabebereichs. Die Adresse ist Pflichtangabe, wenn bei INFO=... die Kurzbezeichnungen SPSPOL, SPSPOLB, SPSPOLC, SPSPOLD, SPSPOLE, SPSPOLF, SPSPOLG, SPSPOLH, SPSPOLI, SPSPOLJ, SPSPOLK, SPSPOLL, SPSPOLM, SPSPOLN, SPSPOLO, SPSPOLP, SPSPOLQ, SPSPOLR, SPSPOLS, SPSPOLT, SPSPOLU, SPSPOLV, SPSPOLW, SPSPOLX, SPSPOLY, SPSPOLZ angegeben werden.
=adr	adr = symbolische Adresse des Ausgabebereichs.
OUTL	bezeichnet die Länge des Ausgabebereichs. Die Mindestlängen sind der nachfolgenden Tabelle zu entnehmen. Die Länge ist Pflichtangabe, wenn bei INFO=... die Kurzbezeichnungen SPSPOL oder SPSPOLB angegeben werden. Bei Angabe von SPSPOL, SPSPOLC, SPSPOLD, SPSPOLE, SPSPOLF, SPSPOLG, SPSPOLH, SPSPOLI, SPSPOLJ, SPSPOLK, SPSPOLL, SPSPOLM, SPSPOLN, SPSPOLO, SPSPOLP, SPSPOLQ, SPSPOLR, SPSPOLS, SPSPOLT, SPSPOLU, SPSPOLV, SPSPOLW, SPSPOLX, SPSPOLY, SPSPOLZ werden die in der Tabelle angegebenen Längen als Voreinstellung benutzt.
	Bei INFO=SPSPOL/SPSPOLB/SPSPOLC/SPSPOLD/SPSPOLE/SPSPOLF/SPSPOLG/SPSPOLH/SPSPOLI/SPSPOLJ/SPSPOLK/SPSPOLL/SPSPOLM/SPSPOLN/SPSPOLO/SPSPOLP/SPSPOLQ/SPSPOLR/SPSPOLS/SPSPOLT/SPSPOLU/SPSPOLV/SPSPOLW/SPSPOLX/SPSPOLY/SPSPOLZ bezeichnet OUTL den Teil der Einträge, die ausgegeben werden sollen. Bei OUTL=20 werden nur die ersten 20 Byte eines jeden Eintrags ausgegeben.
=länge	länge = Länge des Ausgabebereichs.
BUFFER	bezeichnet die Adresse des Ausgabebereichs, wenn die Kurzbezeichnungen SPSPOLB, SPSPOLC, SPSPOLD, SPSPOLE, SPSPOLF, SPSPOLG, SPSPOLH, SPSPOLI, SPSPOLJ, SPSPOLK, SPSPOLL, SPSPOLM, SPSPOLN, SPSPOLO, SPSPOLP, SPSPOLQ, SPSPOLR, SPSPOLS, SPSPOLT, SPSPOLU, SPSPOLV, SPSPOLW, SPSPOLX, SPSPOLY, SPSPOLZ angegeben werden.
=adr	adr = Adresse des Ausgabebereichs. Der Bereich muss auf Wortgrenze ausgerichtet werden.
BUFL	bezeichnet die Länge des Ausgabebereichs, wenn BUFFER=... spezifiziert wird.
=länge	länge = Länge des Ausgabebereichs. Werden BUFFER und BUFL verwendet, sind meist mehrere Aufrufe erforderlich, um alle Einträge zu erhalten. Am Ende der Bearbeitung wird der Name des letzten ausgegebenen Eintrags in der Parameterliste gespeichert und kann als Startpunkt für den nächsten Aufruf verwendet werden. Vor dem ersten Aufruf müssen Sie sicherstellen, dass dieses Feld in der Parameterliste mit Null oder Leerzeichen vorbelegt ist. Der Name dieses Feldes ist &P.KEY. Beim letzten Aufruf ist der Puffer meist nicht voll; jedenfalls ist immer eindeutig festzustellen, ob alle Einträge gelesen wurden. Machen Sie einen Aufruf zuviel, erhalten Sie den Returncode X'08'.

MF	bezeichnet die Makroaufrufform; siehe auch Handbuch „ <a href="#">Makroaufrufe an den Ablaufteil</a> “. Voreinstellung: Standardform, d.h. Befehls- teil und Datenbereich mit der Operandenliste werden generiert.
=E	E-Form: nur der Befehls- teil wird generiert.
=(E,adr)	adr = Adresse der Operandenliste. Voreinstellung: Die Adresse muss in Register R1 übergeben werden.
=L	L-Form: ein Datenbereich mit der Operandenliste wird generiert.
=(L,p)	Die Liste enthält die im Aufruf angegebene Operandenwerte sowie symbolische Namen und erläuternde Equates. Bei Aufruf des Makros SPSxxxD wird der Ausgabebereich generiert. p = Präfix für die symbolischen Namen; Voreinstellung: p=SPP.
=C	C-Form: ein Datenbereich mit der Operandenliste wird generiert.
=(C,p)	Im Aufruf angegebene Operandenwerte werden nicht eingetragen. Der Datenbereich enthält symb. Namen und erläuternde Equates. Bei Aufruf des Makros SPSxxxD wird der Ausgabebereich generiert. p = Präfix für die symbolischen Namen; Voreinstellung: p=SPP.
=D	D-Form: eine Dummy Section (DSECT) zu der Operandenliste oder dem Ausgabebereich wird generiert.
=(D,p)	p = Präfix für die symbolischen Namen in der DSECT; Voreinstellung: p=SPP.

**Kurzbezeichnung für Operanden INFO, Mindestlängen und Erläuterungen**

Kurzbezeichnung		Länge std/min	DSECT-Makro		Ausgabe
Name	Ziffer		Makro	SPSPL	
SPSGEN	( 1)	63 / 16	SPSGEND	SPG	SPOOL-globale Voreinstellungen
SPSDVC	( 2)	112 / 16	SPSDVCD	SPD	DVC-Eintrag für ein Gerät
SPSCHR	( 3)	60 / 16	SPSCHRD	SPC	CHR-Eintrag für einen Zeichensatz
SPSFRM	( 4)	64 / 17	SPSFRMD	SPF	FRM-Eintrag für ein Formular
SPSPOL	( 5)	24		SPPOL	POL-Eintrag für einen Pool
SPSDVP	( 6)	24		SPDVP	DVP-Eintrag für einen gerätespez. Pool
SPSDVCL	(12)	intern	SPSDVCLD	SPDL	Liste (Untermenge) der Gerätenamen
SPSCHRL	(13)	intern	SPSCHRLD	SPCL	Liste (Untermenge) der Zeichensatznamen
SPSFRML	(14)	intern	SPSFRMLD	SPFL	Liste (Untermenge) der Formularnamen
SPSDLB	(15)	28	SPSDVCLD	SPDL	Untermenge der Gerätenamen
SPSCLB	(16)	28	SPSCHRLD	SPCL	Untermenge der Zeichensatznamen
SPSFLB	(17)	29	SPSFRMLD	SPFL	Untermenge der Formularnamen
SPSPOLB	(18)	10		SPPOL	Untermenge der Poolnamen
SPSDPLB	(19)	10		SPDVP	Untermenge der gerätespez. Poolnamen
SPSDVCA	(22)	intern	SPSDVCAD	SPDA	Liste (Untermenge) der DEV-Einträge
SPSCHRA	(23)	intern	SPSCHRAD	SPCA	Liste (Untermenge) der CHR-Einträge
SPSFRMA	(24)	intern	SPSFRMAD	SPFA	Liste (Untermenge) der FRM-Einträge
SPSDAB	(25)	1)	SPSDVCAD	SPDA	Liste (Untermenge) der DEV-Einträge
SPSCAB	(26)	1)	SPSCHRAD	SPCA	Liste (Untermenge) der CHR-Einträge
SPSFAB	(27)	1)	SPSFRMAD	SPFA	Liste (Untermenge) der FRM-Einträge
SPSPOAB	(28)	2)		SPPOL	Liste (Untermenge) der POL-Einträge
SPSDPAB	(29)	2)		SPDVP	Liste (Untermenge) der DVP-Einträge
SPSDVCC	(32)				Anzahl der DEV-Einträge
SPSCHRC	(33)				Anzahl der CHR-Einträge
SPSFRMC	(34)				Anzahl der FRM-Einträge
SPSPOLC	(35)				Anzahl der POL-Einträge
SPSDVPC	(36)				Anzahl der DVP-Einträge
SPSRELM	gibt den intern angeforderten Speicherplatz frei.				

**Erläuterung der Tabellenspalten**

*Kurzbezeichnung:*

Kurzbezeichnung für die gewünschte Information. Der Name oder die Zahl (in Klammern) werden bei INFO=.... angegeben.

**Länge: std/min**

Voreinstellung und minimale Länge des Ausgabebereichs (OUTL oder BUFL); wenn nicht angegeben, muss die Länge mit Hilfe der den Ausgabebereich beschreibenden DSECT ermittelt werden. „intern“ bedeutet, dass der Speicherbereich durch die Makrofunktion angefordert wird. Adresse und Länge des Speicherbereichs werden in den Feldern BUFA und BUFL der Operandenliste zurückgegeben. Durch einen nachfolgenden SPSINF-Aufruf mit INFO=SPSRELM wird der Speicherbereich wieder freigegeben.

1. Minimalwert für BUFL: OUTL + 20;  
Minimalwert für OUTL: 8;
2. Minimalwert für BUFL: OUTL;  
Minimalwert für OUTL: 8.

**Makro:**

Diese Makroaufrufe liefern Beschreibungen der Ausgabebereiche als DSECT oder Datenbereich für den links stehenden INFO-Wert.

**SPSPL:**

Der Makro SPSPL liefert Beschreibungen der Ausgabebereiche als Datenbereich für den links stehenden INFO-Wert. Die Angaben in der Spalte unter SPSPL bezeichnen die Operandenwerte für den Operanden PL=...

**Ausgabe:**

Kurzbeschreibung der ausgegebenen Information. Eine Untermenge wird immer dann ausgegeben, wenn bei ITEM=.... ein Name mit Musterelementen angegeben wurde. Die Anzahl der DEV-/CHR-/FRM-Einträge wird in das Feld COUNT der Operandenliste ausgegeben.

**Returncodes**

Über die Makroausführung wird sowohl im Register R15 (linksbündig) als auch im Feld RETC der Operandenliste ein Returncode übergeben.

<b>X' aa'</b>	<b>Erläuterung</b>
X' 00'	Normale Ausführung.
X' 04'	Ungültiger Operand oder fehlerhafte Operandenliste.
X' 08'	Eintrag bzw. Einträge in der SPOOL-Parameterdatei nicht gefunden.
X' 0C'	Kein Speicherplatz für Ausgabe einer Liste.
X' 10'	Systemfehler.

*Beispiel*

Die in der SPOOL-Parameterdatei festgelegten Voreinstellungen sollen ausgegeben werden. Der Makro SPSGEND erzeugt eine Beschreibung des Ausgabebereichs.

```

BEISPIEL START
PRINT NOGEN
BALR 3,0
USING *,3
SPSINF MF=(E,SPSINF) _____ (01)
TERM      TERM
          DS    OF
SPSINF    SPSINF INFO=SPSGEN,OUTPL=OUTADR1,OUTL=64,MF=L _____ (02)
OUTADR1   DS    CL64
          END

(IN)      START-PROGRAM $ASSEMBH _____ (03)
(OUT)    % ASS6010 V 1.1A00 OF BS2000 ASSEMBH READY
(IN)     COMPILE SOURCE=BEISPIEL,MACRO-LIBRARY=$RZV110.GCLIB.UR.V11.OS10.921012,LISTING=P
( )      ARAMETERS(OUTPUT=LST.ASSEMBH),TEST-SUPPORT=YES
(OUT)    % ASS6011 ASSEMBLY TIME: 1302 MSEC
(OUT)    % ASS6018 0 FLAGS, 0 PRIVILEGED FLAGS, 0 MNOTES
(OUT)    % ASS6019 HIGHEST ERROR-WEIGHT: NO ERRORS
(OUT)    % ASS6006 LISTING GENERATOR TIME: 618 MSEC
(IN)     END
(OUT)    % ASS6012 END OF ASSEMBH
(IN)     LOAD-PROG *MOD(*OMF),TEST=AID _____ (04)
(OUT)    % BLS0001 ### DBL VERSION 119 RUNNING
(OUT)    % BLS0517 MODULE 'BEISPIEL' LOADED
(IN)     %in term
(IN)     %r
(OUT)
(OUT)    STOPPED AT LABEL: TERM , SRC_REF: 21 , SOURCE: BEISPIEL , PROC BEISPIEL
(IN)     %d outadr1%x _____ (05)
(OUT)    ** ITN: #'000B01BF' *** TSN: 1GY6 *****
(NL)     CURRENT PC: 00000018 CSECT: BEISPIEL *****
(NL)     V'0000006C' = OUTADR1 + #'00000000'
(NL)     0000006C (00000000) 00000027 C7C5D540 F0F24BF7 C1F9F100 ....GEN 02.7..
(NL)     0000007C (00000010) 01010201 88020001 40404040 40404040 ....h...
(NL)     0000008C (00000020) 020A0400 00C8000A 0064000A 01000101 .....H.....
(NL)     0000009C (00000030) 0032C8C8 14360201 02010000 40010201 ..HH.....

```

- (01) Aufruf des Makros SPSINF in der E-Form.
- (02) Aufruf des Makros SPSINF in der L-Form. Die für Drucker, Zeichensätze, Formulare, SPOOL-OUT-Optionen, ....., festgelegten Voreinstellungen sollen ausgegeben werden.
- (03) Das (Beispiel-)Programm BEISPIEL wird übersetzt.
- (04) BEISPIEL wird geladen und gestartet. Der Ausgabebereich wird mit Hilfe der Testhilfe AID angezeigt.

(05) In der Ausgabe bedeuten (s. auch nachfolgende Beschreibung des Ausgabebereichs):

```

Version           : 39           (Byte 0-3)
Info-Typ          : GEN          (Byte 4-7)
SPOOL-Version     : 02.7        (Byte 8-14)
SPACE-AFTER-HEADER : YES        (Byte 16 = X'01')
FAMILY-MEMBER-HEADER: NO        (Byte 17 = X'02')
COPY-HEADER       : NO          (Byte 18 = X'02')
CHARACTER-SET     : FILE        (Byte 19 = X'02')
PRINTER SIZE      : 136         (Byte 20 = X'88')
LOCK DEFAULT      : NO          (Byte 21 = X'02')
u.s.w

```

### Layout des Ausgabebereichs (DSECT für den GEN-Eintrag)

LABEL1	SPSGEND		
1 LABEL1	DS	OF	
1	SPSPL	PL=SPG,	C
1		P=SPG	
2 SPGOUT	DS	OF	
2 SPGVER	DS	F	VERSION
2 SPGINFO	DS	CL4'GEN'	INFO TYPE
2 SPGSPPOOL	DS	CL7	SPOOL VERSION
2	DS	OH	FOR ALIGNEMENT
2 SPGHSPAC	DS	X	SPACE AFTER HEADER
2 SPGYES	EQU	1	YES
2 SPGNO	EQU	2	NO
2 SPGHFAM	DS	X	FAMILY MEMBER HEADER
2 * SPGYES	EQU	1	YES
2 * SPGNO	EQU	2	NO
2 SPGHCOP	DS	X	COPY HEADER
2 * SPGYES	EQU	1	YES
2 * SPGNO	EQU	2	NO
2 SPGHCHR	DS	X	CHARACTER SET
2 SPGDEF	EQU	1	DEFAULT
2 SPGFILE	EQU	2	FILE
2 SPGPSIZE	DS	X	/PRINT PRINTER SIZE
2 SPG132	EQU	132	132
2 SPG136	EQU	136	136
2 SPG160	EQU	160	160
2 SPGPLOCK	DS	X	/PRINT LOCK DEFAULT
2 * SPGYES	EQU	1	YES
2 * SPGNO	EQU	2	NO
2 SPGPSHIF	DS	FL1	/PRINT SHIFT DEFAULT
2 SPGPDES	DS	X	/PRINT DEFAULT DESTINATION
2 SPGLOC	EQU	1	LOCAL
2 SPGREM	EQU	2	REMOTE
2 SPGPOL	EQU	3	POOL



2	SPGPOLR	EQU	3		POOL REMOTE	332
2	SPGPOLL	EQU	4		POOL LOCAL	332
2	SPGPDVC	DS	CL8		/PRINT DEVICE NAME (IF PDES=SPGREM)	
2		DS	0H		FOR ALIGNMENT	
2	* SPGYES	EQU	1		YES	
2	* SPGNO	EQU	2		NO	
2	SPGSPUSC	DS	FL1		SPOOLOUT PUNCH SEPARATOR CARDS	
2	SPGRSOBS	DS	X		RSO BUFFER SIZE	
2	SPGRBPLI	DS	X		RBP_STATION_LIMIT	255
2	SPGRSOLI	DS	H		RSO LIMIT	
2	SPGSCKLI	DS	H		CHECKPOINT FOR LINE PRINTERS	
2	SPGSCKPA	DS	H		PAGE PRINTERS	
2	SPGSCKRE	DS	H		REMOTE PRINTERS	
2	SPGSTUNI	DS	X		/STATUS SIZE UNIT	
2	SPGPP	EQU	1		PAM PAGES	
2	SPGLI	EQU	2		LINES	
2	SPGSTLPB	DS	X		/STATUS LINES FACTOR	
2	SPGFAMILY	DS	X		FAMILY (PRINT -> SPOOL 2.5A)	309
2	SPFMYES	EQU	1		YES	315
2	SPFMNO	EQU	2		NO	315
2	* SPECIAL FIELDS			INTRODUCED IN SPOOL V2.2BB		251
2	* SPGYES	EQU	1			250
2	* SPGNO	EQU	2			250
2	SPGSTRSP	DS	H		START-SPOOL-LOW-VAL	251
2	SPGPRRSP	DS	X		PRIORITY REPLAY SESSION	253
2	SPGPRFPP	DS	X		PRIORITY FROM POSITIONING	253
2	SPGRMTLI	DS	FL1		REMOTE-MESSGE-TASK	302
2	SPGMLPI	DS	X		MIN_LINES_PER_PAGE	255
2	* SPECIAL FIELDS			INTRODUCED IN SPOOL 2.3B		300
2	SPGDND	DS	X		DEVICE-TYPE ND	300
2	SPGDSD	DS	X		SD	300
2	SPGDLP	DS	X		LP	300
2	SPGDHP	DS	X		HP	300
2	SPGHFOB	DS	H		FORM-OVERLAY-BUFFER	300
2	SPGHCS	DS	X		CHARACTER-SET-NUMBER	300
2	SPGHROT	DS	X		ROTATION	300
2	SPGEMPTF	DS	X		ACCEPT EMPTY FILE (-> PRINT CMD)	311
2	SPEFYES	EQU	1		YES	315
2	SPEFNO	EQU	2		NO	315
2	SPGTRUNC	DS	X		TRUNCATION (-> PRINT CMD)	311
2	SPFIGN	EQU	1		IGNORE	311
2	SPFKEEP	EQU	2		KEEP FILE	311
2	SPGCLSCH	DS	8X		CURRENT LOCAL SCHED PARAM	306
2	SPGCLPRI	DS	8X		CURRENT LOCAL PRIO. PARAM	306
2	SPGCLWEI	DS	0XL16		CURRENT LOCAL DVC WEIGTH	306
2	SPGCLDFO	DS	X		CURRENT LOCAL DVC WEIGTH FORM	306
2	SPGCLDDI	DS	X		CURRENT LOCAL DVC WEIGTH DIA	306
2	SPGCLDFB	DS	X		CURRENT LOCAL DVC WEIGTH FOB NAME	306
2	SPGCLDCH	DS	X		CURRENT LOCAL DVC WEIGTH CHAR-SET	306
2	SPGCLDRO	DS	X		CURRENT LOCAL DVC WEIGTH ROTATION	306

2	SPGCLDFS	DS	X	CURRENT	LOCAL	DVC	WEIGHT	FORM-SIZE	306
2	SPGCLDC#	DS	X	CURRENT	LOCAL	DVC	WEIGHT	FOB CHR NUM	306
2	SPGCLDPR	DS	X	CURRENT	LOCAL	DVC	WEIGHT	PRIORITY	306
2	SPGCLDUS	DS	X	CURRENT	LOCAL	DVC	WEIGHT	USERID	306
2	SPGCLDCL	DS	X	CURRENT	LOCAL	DVC	WEIGHT	CLASS	306
2	SPGCLDFI	DS	X	CURRENT	LOCAL	DVC	WEIGHT	FILE-SIZE	306
2	SPGCLDPN	DS	X	CURRENT	LOCAL	DVC	WEIGHT	PNAME	306
2	SPGCLDAC	DS	X	CURRENT	LOCAL	DVC	WEIGHT	ACCOUNT	306
2	SPGCLDAG	DS	X	CURRENT	LOCAL	DVC	WEIGHT	AGE	306
2	SPGCLDSQ	DS	X	CURRENT	LOCAL	DVC	WEIGHT	SEQUENCE	101
2	SPGCLDRS	DS	X	CURRENT	LOCAL	DVC	WEIGHT	RESERVED	331
2	SPGCRSCH	DS	8X	CURRENT	RSO	SCHED	PARAM		306
2	SPGCRPRI	DS	8X	CURRENT	RSO	PRIO.	PARAM		306
2	SPGCRWEI	DS	0XL16	CURRENT	RSO	DVC	WEIGHT		306
2	SPGCRDFO	DS	X	CURRENT	RSO	DVC	WEIGHT	FORM	306
2	SPGCRDDI	DS	X	CURRENT	RSO	DVC	WEIGHT	DIA	306
2	SPGCRDFB	DS	X	CURRENT	RSO	DVC	WEIGHT	FOB NAME	306
2	SPGCRDCH	DS	X	CURRENT	RSO	DVC	WEIGHT	CHAR-SET	306
2	SPGCRDRO	DS	X	CURRENT	RSO	DVC	WEIGHT	ROTATION	306
2	SPGCRDFS	DS	X	CURRENT	RSO	DVC	WEIGHT	FORM-SIZE	306
2	SPGCRDC#	DS	X	CURRENT	RSO	DVC	WEIGHT	FOB CHR NUM	306
2	SPGCRDPR	DS	X	CURRENT	RSO	DVC	WEIGHT	PRIORITY	306
2	SPGCRDUS	DS	X	CURRENT	RSO	DVC	WEIGHT	USERID	306
2	SPGCRDCL	DS	X	CURRENT	RSO	DVC	WEIGHT	CLASS	306
2	SPGCRDFI	DS	X	CURRENT	RSO	DVC	WEIGHT	FILE-SIZE	306
2	SPGCRDPN	DS	X	CURRENT	RSO	DVC	WEIGHT	PNAME	306
2	SPGCRDAC	DS	X	CURRENT	RSO	DVC	WEIGHT	ACCOUNT	306
2	SPGCRDAG	DS	X	CURRENT	RSO	DVC	WEIGHT	AGE	306
2	SPGCRDSQ	DS	X	CURRENT	RSO	DVC	WEIGHT	SEQUENCE	101
2	SPGCRDRS	DS	X	CURRENT	RSO	DVC	WEIGHT	RESERVED	306
2	SPGNLSCH	DS	8X	NEXT	LOCAL	SCHED	PARAM		306
2	SPGNLPRI	DS	8X	NEXT	LOCAL	PRIO.	PARAM		306
2	SPGNLWEI	DS	0XL16	NEXT	LOCAL	DVC	WEIGHT		306
2	SPGNLDFO	DS	X	NEXT	LOCAL	DVC	WEIGHT	FORM	306
2	SPGNLDDI	DS	X	NEXT	LOCAL	DVC	WEIGHT	DIA	306
2	SPGNLDFB	DS	X	NEXT	LOCAL	DVC	WEIGHT	FOB NAME	306
2	SPGNLDCH	DS	X	NEXT	LOCAL	DVC	WEIGHT	CHAR-SET	306
2	SPGNLDRO	DS	X	NEXT	LOCAL	DVC	WEIGHT	ROTATION	306
2	SPGNLDFS	DS	X	NEXT	LOCAL	DVC	WEIGHT	FORM-SIZE	306
2	SPGNLDC#	DS	X	NEXT	LOCAL	DVC	WEIGHT	FOB CHR NUM	306
2	SPGNLDPR	DS	X	NEXT	LOCAL	DVC	WEIGHT	PRIORITY	306
2	SPGNLDUS	DS	X	NEXT	LOCAL	DVC	WEIGHT	USERID	306
2	SPGNLDCL	DS	X	NEXT	LOCAL	DVC	WEIGHT	CLASS	306
2	SPGNLDFI	DS	X	NEXT	LOCAL	DVC	WEIGHT	FILE-SIZE	306
2	SPGNLDPN	DS	X	NEXT	LOCAL	DVC	WEIGHT	PNAME	306
2	SPGNLDAC	DS	X	NEXT	LOCAL	DVC	WEIGHT	ACCOUNT	306
2	SPGNLDAG	DS	X	NEXT	LOCAL	DVC	WEIGHT	AGE	306
2	SPGNLDSQ	DS	X	NEXT	LOCAL	DVC	WEIGHT	SEQUENCE	101
2	SPGNLDRE	DS	X	NEXT	LOCAL	DVC	WEIGHT	RESERVED	306
2	SPGNRSCH	DS	8X	NEXT	RSO	SCHED	PARAM		306

2	SPGNRPRI	DS	8X	NEXT	RSO	PRIO. PARAM	306
2	SPGNRWEI	DS	0XL16	NEXT	RSO	DVC WEIGHT	306
2	SPGNRDFO	DS	X	NEXT	RSO	DVC WEIGHT FORM	306
2	SPGNRDDI	DS	X	NEXT	RSO	DVC WEIGHT DIA	306
2	SPGNRDFB	DS	X	NEXT	RSO	DVC WEIGHT FOB NAME	306
2	SPGNRDCH	DS	X	NEXT	RSO	DVC WEIGHT CHAR-SET	306
2	SPGNRDRO	DS	X	NEXT	RSO	DVC WEIGHT ROTATION	306
2	SPGNRDFS	DS	X	NEXT	RSO	DVC WEIGHT FORM-SIZE	306
2	SPGNRDC#	DS	X	NEXT	RSO	DVC WEIGHT FOB CHR NUM	306
2	SPGNRDPR	DS	X	NEXT	RSO	DVC WEIGHT PRIORITY	306
2	SPGNRDUS	DS	X	NEXT	RSO	DVC WEIGHT USERID	306
2	SPGNRDCL	DS	X	NEXT	RSO	DVC WEIGHT CLASS	306
2	SPGNRDFI	DS	X	NEXT	RSO	DVC WEIGHT FILE-SIZE	306
2	SPGNRDPN	DS	X	NEXT	RSO	DVC WEIGHT PNAME	306
2	SPGNRDAC	DS	X	NEXT	RSO	DVC WEIGHT ACCOUNT	306
2	SPGNRDAG	DS	X	NEXT	RSO	DVC WEIGHT AGE	306
2	SPGNRDSQ	DS	X	NEXT	RSO	DVC WEIGHT SEQUENCE	101
2	SPGNRDRE	DS	X	NEXT	RSO	DVC WEIGHT RESERVED	306
2	SPGPOOL	DS	CL8			POOL NAME	306
2	SPGCTR#	DS	H			MAX NUMBER OF CONTROLLER	306
2	SPGDVC#	DS	H			MAX NUMBER OF DEVICE/CNTRL	306
2	SPGCHKTI	DS	X			CHECK TIME VALUE	306
2	SPGACCNB	DS	X			PRINT ACCOUNT NUMBER (PRINT CMD)	316
2	SPACYES	EQU	1			YES	315
2	SPACNO	EQU	2			NO	315
2	SPGRELMM	DS	X			RELEASE MEMORY (YES-NO)	306
2	SPGFNMLG	DS	X			FORM NAME LENGTH (SD CMD)	316
2	SPGRPTIM	DS	H			REPEAT TIME	306
2	SPGMLPID	DS	X			MIN L-P-P INDIC	318
2	SPIDNSP	EQU	0			NOT SPECIFIED	318
2	SPIDSP	EQU	1			SPECIFIED	318
2	SPGSCMAX	DS	X			# MAXIMUM OF SECTION CTRL	319
2	SPGND50L	DS	X			DEVICE-TYPE ND50-LINE-PRINTER	330
2	SPMND50Y	EQU	1			YES	330
2	SPMND50N	EQU	2			NO	330
2	SPGRSOFM	DS	X			RSO FAMILY PROCESSING	330
2	SPMRFMY	EQU	1			YES	330
2	SPMRFMN	EQU	2			NO	330
2	SPGLSTFR	DS	CL8			DEFAULT SYSLST FORM	
2	SPGOUTFR	DS	CL8			DEFAULT SYSOUT FORM	
2	SPGNDX	DS	X			DEVICE-TYPE NDX-PRINTER	330
2	SPMNDXY	EQU	1			YES	330
2	SPMNDXN	EQU	2			NO	330
2	SPGNDXFO	DS	X			NDX-PRINTER : FORMS OVERLAY	330
2	SPMFOY	EQU	1			YES	330
2	SPMFON	EQU	2			NO	330
2	SPGRPM	DS	X			NDX-PRINTER : RASTER PATTERN MEMORY	330
2	SPM18	EQU	1			1800	330
2	SPM36	EQU	2			3600	330
2	SPM20	EQU	4			2048	

2	SPM40	EQU	8	4096	
2	SPM8192	EQU	16	8192	406
2	SPM12288	EQU	32	12288	406
2	SPM16384	EQU	64	16384	406
2	SPGNXROT	DS	X	DEVICE-TYPE APA-PRINTER	406
2	SPMNXRY	EQU	1	YES	406
2	SPMNXRN	EQU	2	NO	406
2	SPGNXFOB	DS	H	NDX-PRINTER : FORM-OVERLAY-BUFFER	330
2	SPGNXCSN	DS	X	NDX-PRINTER : CHARACTER-SET-NUMBER	330
2	SPGPRSCP	DS	X	PRIORITY SECTION POSITIONING	339
2	SPGAPA	DS	X	DEVICE-TYPE APA-PRINTER	400
2	SPMAPAY	EQU	1	YES	400
2	SPMAPAN	EQU	2	NO	400
2	SPGTWIN	DS	X	DEVICE-TYPE TWIN-PRINTER	*620
2	SPMGTWY	EQU	1	YES	*620
2	SPMGTWN	EQU	2	NO	*620
2	SPGRCRP	DS	CL38	RECOVERY-RULES-PREFIX	*625
2	SPGDDST	DS	AL1	DPRINT DESTINATION	*643
2	SPGDPOL	DS	CL8	DPRINT DEFAULT POOL	*643
2	SPGDRSL	DS	AL1	RESOURCE LOCATION	*643
2	SPGDSTL	DS	AL1	SERVER TASK LIMIT	*630
2	SPGDRTL	DS	AL1	RESOURCE TASK LIMIT	*630
2	SPGDRJP	DS	AL1	REMOTE JOB PRIORITY	*630
2		DS	X	RESERVED	*645
2	SPGDTAD	DS	XL32	TRANSFER ADMISSION	*630
2	SPGDOCT	DS	H	MAX LOGICAL MSG LENGTH	*645
2	SPGTALN	DS	H	TRANSFER ADMISSION LENGTH	*646
2	SPGTATP	DS	X	TRANSFER ADMISSION SDF TYPE	*646
2	SPGGRSD	DS	34X	RESERVED	645
2	SPGLEN	EQU	*-SPGOUT		
	OUTADR1	DS	CL64		
		END			

---

## 4 Makros für die Verwaltung von Druckaufträgen

Für die Verwaltung von Druckaufträgen wird eine Programmierschnittstelle angeboten, die vier Makros umfasst. Diese Makros bieten die gleiche Funktionalität wie die entsprechenden Kommandos.

- [„CANPJOB - Druckauftrag abbrechen“ auf Seite 215](#) (Kommando CANCEL-PRINT-JOB)  
Mit diesem Makro wird ein Druckauftrag abgebrochen.
- [„MODPJAT - Attribute eines Druckauftrags modifizieren“ auf Seite 220](#) (Kommando MODIFY-PRINT-JOB-ATTRIBUTES)  
Mit diesem Makro werden die Attribute eines Druckauftrags verändert.
- [„SHOPJAT - Attribute eines Druckauftrags anzeigen“ auf Seite 297](#) (Kommando SHOW-PRINT-JOB-ATTRIBUTES)  
Mit diesem Makro werden die Attribute eines Druckauftrags ausgegeben.
- [„SHOPJST - Status eines Druckauftrags anzeigen“ auf Seite 305](#) (Kommando SHOW-PRINT-JOB-STATUS)  
Mit diesem Makro wird der Status eines Druckauftrags ausgegeben.

Im [Abschnitt „Unterstützende Makros für MODPJAT“ auf Seite 356](#) werden die folgenden Makros beschrieben:

- [„MODJRSO - RSO-Parameter für MODPJAT-Makro festlegen“ auf Seite 357](#)
- [„MPJADPC - Dprint-Parameter für MODPJAT-Makro definieren“ auf Seite 364](#)
- [„MPJASPS - SPS-Parameter für MODPJAT-Makro definieren“ auf Seite 366](#)

Im [Abschnitt „Ausgabestrukturen des SHOPJAT-Makros“ auf Seite 371](#) werden die folgenden Ausgabestrukturen beschrieben:

- [„SJAODOF - Ausgabestruktur für INFO=\\*DOCUMENT\\_FORMAT“ auf Seite 372](#)
- [„SJAODOP - Ausgabestruktur für INFO=\\*DOCUMENT\\_PART“ auf Seite 374](#)
- [„SJAOLCT - Ausgabestruktur für INFO=\\*LAYOUT\\_CONTROL“ auf Seite 377](#)
- [„SJAOPJC - Ausgabestruktur für INFO=\\*PRINT\\_JOB\\_CONTROL“ auf Seite 380](#)
- [„SJAORSD - Ausgabestruktur für INFO=\\*RESOURCES\\_DESCRIPTION“ auf Seite 381](#)
- [„SJAOTOP - Ausgabestruktur für INFO=\\*TO\\_PRINTER“ auf Seite 383](#)
- [„SPJARSO - RSO-Ausgabestruktur für SHOPJAT-Makro“ auf Seite 384](#)
- [„SPJADPC - Dprint-Ausgabestruktur für SHOPJAT-Makro“ auf Seite 385](#)
- [„SPJASPS - SPS-Ausgabestruktur für SHOPJAT-Makro“ auf Seite 386](#)

Im Abschnitt „Unterstützende Makros für MODPJAT“ auf Seite 356 ist der folgende Makro beschrieben:

- „SPJSRSO - RSO-Parameter für SHOPJST-Makro festlegen“ auf Seite 387

Im Abschnitt „Ausgabestrukturen des SHOPJST-Makros“ auf Seite 390 werden die folgenden Ausgabestrukturen beschrieben:

- „SJSODES - Ausgabestruktur für DESTINATION“ auf Seite 391
- „SJSODES - Ausgabestruktur für SPOOL-FILTER“ auf Seite 394
- „SJSODES - Ausgabestruktur für RSO-FILTER“ auf Seite 397
- „SJSODIS - Ausgabestruktur für DISTRIBUTED“ auf Seite 400
- „SJSODSS - Ausgabestruktur für DESTINATION und UNIX-Systeme“ auf Seite 402
- „SJSOORI - Ausgabestruktur für ORIGIN“ auf Seite 404
- „SJSOORS - Ausgabestruktur für ORIGIN und UNIX-Systeme“ auf Seite 406
- „SJSOTRA - Ausgabestruktur für TRAITS“ auf Seite 407
- „SJSOTRS - Ausgabestruktur für TRAITS und UNIX-Systeme“ auf Seite 409

## CANPJOB - Druckauftrag abbrechen

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprachen:** Assembler, C, COBOL

**Makrotyp:** C, D, E, L, M

Mit diesem Makro wird ein Druckauftrag abgebrochen. Das Makro entspricht dem Kommando CANCEL-PRINT-JOB.

### Format

Operation	Operanden
CANPJOB	<p>VARIANT=<u>001</u> / &lt;c-string 3..3&gt;</p> <p>,TSN=(<i>tsn1</i>, <i>cluste1</i>)  <i>tsn1</i>: &lt;var: char: 4&gt; / &lt;c-string 4..4: alphanum-name_4..4&gt;  <i>cluste1</i>: <u>*LOCAL CLUSTER</u> / &lt;var: char: 8&gt; / &lt;c-string 1..8: name_1..8&gt;</p> <p>SERVTSN=(<i>tsn2</i>, <i>server</i>)  <i>tsn2</i>: &lt;var: char: 4&gt; / &lt;c-string 4..4: alphanum-name_4..4&gt;  <i>server</i>: &lt;var: char: 8&gt; / &lt;c-string 1..8: name_1..8&gt;</p> <p>MONJV=<u>*NONE</u> / &lt;var: char: 54&gt; /  &lt;c-string 1..54: filename_1..54_with-catid&gt;</p> <p>FOREIGN=(<i>identif</i>, <i>cluste2</i>):  <i>identif</i>: <u>*NONE</u> / &lt;var: int: 4&gt; / &lt;integer 1..2147483647&gt;  <i>cluste2</i>: &lt;var: char: 8&gt; / &lt;c-string 1..8: name_1..8&gt;</p> <p>SYSFILE=array(16): <u>*DUMMY</u> / &lt;var :int: 1&gt; / *SYSOUT / *SYSLST /  &lt;integer0..99&gt;</p>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

#### Hinweis

Die Operanden TSN, SERVTSN, MONJV, FOREIGN und SYSFILE schließen sich gegenseitig aus. Daher darf von diesen Operanden immer nur einer verwendet werden.

**TSN=(tsn1, cluste1)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Client-Rechner.

**tsn1:** <var: char: 4> / <c-string 4..4: alphanum-name\_4..4>

TSN des Druckauftrags, der abgebrochen werden soll.

**tsn1:** <var: char: 4>

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**tsn1:** <c-string 4..4: alphanum-name\_4..4>

Die TSN des Druckauftrags wird direkt angegeben.

**cluste1:** \*LOCAL\_CLUSTER / <var: char: 8> / <c-string 1..8: name\_1..8>

Name des Clusters, in dem der Druckauftrag abgearbeitet werden sollte. Wenn ein Name angegeben wurde, bezieht sich die TSN auf den Gateway-Host im angegebenen Cluster. Es können nur BS2000-Cluster angegeben werden.

**cluste1:** \*LOCAL\_CLUSTER

Vorgabewert für den Namen des Clusters, in dem der Druckauftrag abgearbeitet werden sollte.

**cluste1:** <var: char: 8>

Der Name des Clusters wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**cluste1:** <c-string 1..8: name\_1..8>

Der Name des Clusters wird direkt angegeben.

**SERVTSN=(tsn2, server)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Server-Rechner. Nur Druckaufträge im lokalen Cluster können auf diese Weise angesprochen werden.

**tsn2:** <var: char: 4> / <c-string 4..4: alphanum-name\_4..4>

TSN auf dem Server-Rechner des Druckauftrags, der abgebrochen werden soll.

**tsn2:** <var: char: 4>

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**tsn2:** <c-string 4..4: alphanum-name\_4..4>

Die TSN des Druckauftrags wird direkt angegeben.

**server:** <var: char: 8> / <c-string 1..8: name\_1..8>

Name des Server-Rechners, auf dem der Druckauftrag abgearbeitet wird.

**server:** <var: char: 8>

Der Name des Server-Rechners wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.



*server*: <c-string 1..8: name\_1..8>

Der Name des Server-Rechners wird direkt angegeben.

**MONJV=\*NONE** / <var: char: 54> / <c-string 1..54: filename\_1..54\_with-catid>

Der Druckauftrag wird über die zu diesem Auftrag gehörende MONJV angesprochen, vorausgesetzt, dass auf die angegebene MONJV auf dem lokalen Rechner zugegriffen werden kann.

**MONJV=\*NONE**

Es wurde keine MONJV angegeben.

**MONJV=<var: char: 54>**

Der Name der MONJV wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**MONJV=<c-string 1..54: filename\_1..54\_with-catid>**

Der Name der MONJV wird direkt angegeben.

**FOREIGN=(*identif*, *cluste2*):**

Der Druckauftrag wird über eine Nicht-BS2000-Identifikation angesprochen.

*identif*: \*NONE / <var: int: 4> / <integer 1..2147483647>

Identifikation des Nicht-BS2000-Verweises auf den Druckauftrag.

*identif*: \*NONE

Vorgabewert, keine Identifikation angegeben.

*identif*: <var: int: 4>

Die Identifikation wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*identif*: <integer 1..2147483647>

Die Identifikation wird direkt angegeben.

*cluste2*: <var: char: 8> / <c-string 1..8: name\_1..8>

Name des Clusters, in dem der Druckauftrag abgearbeitet wird.

*cluste2*: <var: char: 8>

Der Name des Clusters wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*cluste2*: <c-string 1..8: name\_1..8>

Der Name des Clusters wird direkt angegeben.

**SYSFILE=array(16): \*DUMMY** / <var :int: 1> / \*SYSOUT / \*SYSLST / <integer 0..99>

Gibt die Systemdatei an, die verarbeitet werden sollte. Dieser Operand stoppt das von einem vorangegangenen Kommando PRINT-DOCUMENT gestartete START-PROCESSING.

**SYSFILE=\*DUMMY**

Vorgabewert für diesen Operanden.

**SYSFILE=<var :int: 1>**

Die Identifikation der Systemdatei wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**SYSFILE=\*SYSOUT**

**SYSFILE=\*SYSLST**

**SYSFILE=<integer 0..99>**

Die Identifikation der Systemdatei wird direkt angegeben.

### Returncodes

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	kein Fehler
02	00	0000	Druckauftrag nicht gefunden oder Verarbeitung nicht erlaubt
01	01	FFFF	Ungültiger Ungültiger Parameter
02	01	FFFF	ungültige Adresse / Länge
03	01	FFFF	Ausgabebereich zu kurz
00	20	FFFF	Systemfehler
02	40	FFFF	Subcode 1 = 40 : keine Autorisation
04	40	FFFF	Wertefehler im Operanden
05	40	FFFF	P/L-Version nicht unterstützt
00	80	FFFF	Subsystem nicht geladen
01	80	FFFF	SPOOL nicht geladen
02	80	FFFF	RSO nicht geladen
03	80	FFFF	PLAM nicht geladen
04	80	FFFF	JV nicht geladen
05	80	FFFF	SPS nicht geladen
06	80	FFFF	DPRINTCL nicht geladen
07	80	FFFF	DPRINTCM nicht geladen
08	80	FFFF	DPRINTSV nicht geladen
09	80	FFFF	DSEM nicht geladen

### Hinweis

Das Makro CANPJOB gibt den Returncode zurück. Zusätzlich können die Returncodes von HDRCHECK und \$VALID zurückgegeben werden.

Wenn kein Datenbereich für den Header bereitgestellt wurde, wird der Wert X'0000003C im Register 15 zurückgegeben.

**Beschreibung der Parameterliste**

```

*   parameter list description
SCPJPL      DS      OF
SCPJHDR     FHDR   MF=(C,SCPJ),EQUATES=NO      Standard header
*   main return codes
SCPJRCNO    EQU    0                          subcode1 = 00 : no error
SCPJWARN    EQU    2                          job not found or processing
*                                                  not permitted
SCPJPARM    EQU    1                          subcode1 = 01 : wrong
*                                                  parameter
SCPJVALD    EQU    2                          invalid address/length
SCPJSHRT    EQU    3                          output area too short
SCPJSYSE    EQU    0                          subcode1 = 20 : system error
SCPJPRIV    EQU    2                          subcode1 = 40 : no
*                                                  authorization
SCPJVERR    EQU    4                          operand value error
SCPJVERE    EQU    5                          p/l version not supported
SCPJNOSS    EQU    0                          subcode1 = 80 : subsystem not
*                                                  loaded
SCPJNSPO    EQU    1                          spool not loaded
SCPJNRSO    EQU    2                          rso not loaded
SCPJNPLM    EQU    3                          plam not loaded
SCPJNJV     EQU    4                          jv not loaded
SCPJNSPS    EQU    5                          sps not loaded
SCPJNDCL    EQU    6                          dprintcl not loaded
SCPJNDCM    EQU    7                          dprintcm not loaded
SCPJNDSV    EQU    8                          dprintsv not loaded
SCPJNDSM    EQU    9                          dsem not loaded
*
SCPJTSN     DS      CL4                       client or server TSN
SCPJCLUS    DS      CL8                       cluster name
SCPJSERV    DS      CL8                       server name
SCPJMJV     DS      CL54                      MONJV name
SCPJSYSFILE DS      16X                      list of system files
SCPJOPT     DS      AL1                      _option
SCPJOTSN    EQU    X'80'                      *TSN
SCPJOSVT    EQU    X'40'                      *SERVER-TSN
SCPJOMJV    EQU    X'20'                      *MONJV
SCPJOFRG    EQU    X'10'                      *FOREIGN
SCPJOSYS    EQU    X'08'                      *SYSFILE
SCPJOFRE    EQU    X'07'                      free bits
SCPJUN2     DS      XL1
SCPJIDEN    DS      F                        print job identification on
*                                                  non-BS2000 cluster
SCPJ#       EQU    *-SCPJHDR

```

## MODPJAT - Attribute eines Druckauftrags modifizieren

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprachen:** Assembler, C, COBOL

**Makrotyp:** C, D, E, L, M

Mit diesem Makro werden die Attribute eines Druckauftrags verändert. Das Makro entspricht dem Kommando MODIFY-PRINT-JOB-ATTRIBUTES.

### Format

Operation	Operanden
MODPJAT	VARIANT= <u>001</u> / <c-string 3..3>  CALLER= <u>*USER</u> / *SYSTEM  RSOPAR= <u>*NONE</u> / <var: pointer>  SPSPAR= <u>*NONE</u> / <var: pointer>  DPPAR= <u>*NONE</u> / <var: pointer>  DSEMPAR= <u>*NONE</u> / <var: pointer>  TSN=( <i>tsn1</i> , <i>cluste1</i> ) <i>tsn1</i> : <var: char: 4> / <c-string 4..4: alphanum-name_4..4> <i>cluste1</i> : <u>*LOCAL_CLUSTER</u> / <var: char: 8> / <c-string 1..8: name_1..8>  SRV_TSN=( <i>tsn2</i> , <i>server</i> ) <i>tsn2</i> : <var: char: 4> / <c-string 4..4> <i>server</i> : <var: char: 8> / <c-string 1..8: name_1..8>  CLT_TSN=( <i>tsn2</i> , <i>client</i> ) <i>tsn2</i> : <var: char: 4> / <c-string 4..4: alphanum-name_4..4> <i>client</i> : <var: char: 8> / <c-string 1..8: name_1..8>

(Teil 1 von 6)

Operation	Operanden
MODPJAT	<p>MONJV=<u>*NONE</u> / &lt;var: char: 54&gt; /  &lt;c-string 1..54: filename_1..54_with-catid&gt;</p> <p>FOREIGN=(<i>identif, cluste2</i>)  <i>identif</i>: <u>*NONE</u> / &lt;var: int: 4&gt; / &lt;integer 1..2147483647&gt; /  *WHOLE-FILE  <i>cluste2</i>: &lt;var: char: 8&gt; / &lt;c-string 1..8: name1..8&gt;  <i>sectyp</i>: <u>*UNCHANGED</u> / *CHAR / &lt;var: bit:1&gt; / *HEXA  <i>sectpos</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..2047&gt; / *STD</p> <p>SECTREC=(<i>sectid, sectlen, secttyp, sectpos</i>)  <i>sectid</i>: <u>*UNCHANGED</u> / &lt;var: char: 60&gt; /  &lt;c-string 1..60: c-string_1..60&gt; / *WHOLE-FILE  <i>sectlen</i>: <u>Q</u> / &lt;var: int: 1&gt; / &lt;integer 1..60&gt;  <i>sectyp</i>: <u>*UNCHANGED</u> / *CHAR / &lt;var: bit:1&gt; / *HEXA  <i>sectpos</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..2047&gt; / *STD</p> <p>FIRSREC=(<i>stpos, ststrid, ststrle, secttyp, ststrpo, ststroc</i>)  <i>stpos</i>: &lt;var: int: 4&gt; / &lt;integer 1..2147483647&gt; / *BEGIN_OF_FILE /  *BY_STRING_ID  <i>ststrid</i>: <u>*UNCHANGED</u> / &lt;var: char: 60&gt; / &lt;c-string 1..60: c-string_1..60&gt;  <i>ststrle</i>: <u>Q</u> / &lt;var: int: 1&gt; / &lt;integer 1..60&gt;  <i>sectyp</i>: <u>*UNCHANGED</u> / *CHAR / &lt;var: bit:1&gt; / *HEXA  <i>ststrpo</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..2047&gt; / *STD  <i>ststroc</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt;</p> <p>LASTREC=(<i>enpos, enstrid, enstrle, secttyp, enstrpo, enstroc</i>)  <i>enpos</i>: &lt;var: int: 4&gt; / &lt;integer 1..2147483647&gt; / *END_OF_FILE /  *BY_STRING_ID  <i>enstrid</i>: <u>*UNCHANGED</u> / &lt;var: char: 60&gt; / &lt;c-string 1..60: c-string_1..60&gt;  <i>enstrle</i>: <u>Q</u> / &lt;var: int: 1&gt; / &lt;integer 1..60&gt;  <i>sectyp</i>: <u>*UNCHANGED</u> / *CHAR / &lt;var: bit:1&gt; / *HEXA  <i>enstrpo</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..2047&gt; / *STD  <i>enstroc</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt;</p> <p>RECPART=(<i>firstch, lastch</i>)  <i>firstch</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt;  <i>lastch</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt;</p>

(Teil 2 von 6)

Operation	Operanden
MODPJAT	<p>OUTPART=(<i>from, to, dim</i>)  <i>from</i>: <u>*UNCHANGED</u> / &lt;var: int: 4&gt; /                    &lt;integer -2147483647..2147483647&gt; / *BEGIN_OF_FILE  <i>to</i>: <u>*UNCHANGED</u> / &lt;var: int: 4&gt; / &lt;integer 1..2147483647&gt; /                *END_OF_FILE  <i>dim</i>: <u>*UNCHANGED</u> / &lt;var: enum-of _layout_part_dim_set: 1&gt; /                *PAGES / *LINES</p> <p>DOCFORM=<u>*UNCHANGED</u> / &lt;var: enum-of _docform_set: 1&gt; /*TEXT /                *PAGE_FORMAT / *SPECIAL_FORMAT</p> <p>LINESP=(<i>spacing, ccpos</i>)  <i>spacing</i>: <u>*UNCHANGED</u> / &lt;var: enum-of _space_set:1&gt; / *SPACE_1 /                    *SPACE_2 / *SPACE_3 / *BY_ASA_CONTROL /                    *BY_EBCDIC_CONTROL / *BY_IBM_CONTROL / *NO  <i>ccpos</i>: <u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..2040&gt; / *STD</p> <p>CONTMOD=(<i>mode, pcc, ctltype</i>)  <i>mode</i>: <u>*UNCHANGED</u> / &lt;var: enum-of _control_mode_set: 1&gt; / *STD /                *PHYSICAL / *LINE_MODE / *LOGICAL / *APA /                *PAGE_MODE  <i>pcc</i>: <u>*UNCHANGED</u> / &lt;var: enum-of _pcc_set:1&gt; / *YES / *NO  <i>ctltype</i>: <u>*UNCHANGED</u> / &lt;var: enum-of _control_type_set:1&gt; /                *COMPATIBLE / *HP</p> <p>ADDCOP=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 0..255&gt;</p> <p>CHECKP=<u>*UNCHANGED</u> / &lt;var: enum-of _checkp_set:1&gt; /                *ON_PAGES / *ON_SECTION_RECORDS</p> <p>PRJNAME=<u>*UNCHANGED</u> / &lt;var: char: 8&gt; /                &lt;c-string 1..8: c-string_with-lower_case_1..8&gt;</p> <p>PRJPRIO=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 30..255&gt;</p> <p>TRUNC=<u>*UNCHANGED</u> / &lt;var: enum-of _truncation_set:1&gt; / *STD /                *DELETE_FILE / *KEEP_FILE</p>

(Teil 3 von 6)

Operation	Operanden
MODPJAT	<p>LOCKF=<u>*UNCHANGED</u> / &lt;var: enum-of _lockf_set:1&gt; / *YES / *NO</p> <p>DELF=<u>*UNCHANGED</u> / &lt;var: enum-of _delf_set: 1&gt; / *NO / *YES / *DESTROY</p> <p>PAGECOP=<u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 0..255&gt; / *STD</p> <p>LEFTMAR=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 0..31&gt; / *STD</p> <p>LINEPP=<u>*UNCHANGED</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt; / *STD</p> <p>HEADLIN=<u>*UNCHANGED</u> / &lt;var: enum-of _headline_set:1&gt; / *NO / *STD / *DATE / *FRECORD / *PAGE / *DATE_FRECORD / *DATE_PAGE / *FRECORD_PAGE / *DATE_FRECORD_PAGE</p> <p>OUTFORM=<u>*UNCHANGED</u> / &lt;var: enum-of _outform_set:1&gt; / *CHARACTER / *HEXADECIMAL</p> <p>TWOSIDE=<u>*UNCHANGED</u> / &lt;var: enum-of _twoside_set:1&gt; / *STD / *NO / *YES / *TUMBLE / *IGNORE</p> <p>ROT=<u>*UNCHANGED</u> / &lt;var: enum-of _rotation_set: 1&gt; / *NO / *ROT_0 / *ROT_90 / *ROT_180 / *ROT_270 / *ROT_0_180 / *ROT_90_270 / *ROT_180_0 / *ROT_270_90 / *BY_CONTROL_CODES</p> <p>INTRAY=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 1..99&gt; / *STD / *IGNORE / *BY_FORMAT</p> <p>INTRAYF=&lt;var: enum-of _inray_format_set:1&gt; / *MANUAL / *A3 / *A4 / *A5 / *B4 / *B5 / *DOUBLE_LETTER / *EXEC / *FOLIO / *INVOICE / *LEGAL / *LETTER / *MONARCH / *COMMERCIAL_10 / *DL / *C5 / *A3_UNCUT / *A4_UNCUT / *LEDGER</p> <p>OUTTRAY=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 1..99&gt; / *STD / *IGNORE / *SORTER</p>

(Teil 4 von 6)

Operation	Operanden
MODPJAT	<p>COVPAGE=(<i>headtxt, headexi, traixi</i>)  <i>headtxt</i>: &lt;var: char: 32&gt; / &lt;c-string 1..32: c-string_1..32&gt; / *NONE  <i>headexi</i>: <u>*UNCHANGED</u> / &lt;var: int: 4&gt; / &lt;integer 0..2147483639&gt; / *NO  <i>traixi</i>: <u>*UNCHANGED</u> / &lt;var: int: 4&gt; / &lt;integer 0..2147483639&gt; / *NO</p> <p>FORM=<u>*UNCHANGED</u> / &lt;var: char: 6&gt; /  &lt;c-string 1..6: alphanum_name_1..6&gt; / *STD</p> <p>LOOP=<u>*UNCHANGED</u> / &lt;var: char: 3&gt; /  &lt;c-string 1..3: alphanum_name_1..3&gt; / *STD</p> <p>ROTLLOOP=<u>*UNCHANGED</u> / &lt;var: char: 3&gt; /  &lt;c-string 1..3: alphanum_name_1..3&gt; / *STD</p> <p>CHARSET=<u>*UNCHANGED</u> / array(16): &lt;var: char: 3&gt; /  &lt;c-string 1..3: alphanum_name_1..3&gt; / *STD</p> <p>POOLNAM=<u>*UNCHANGED</u> / &lt;var: char: 4&gt; / &lt;c-string 1..4: name_1..4&gt; /  *NONE</p> <p>POOLIND=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 0..64&gt;</p> <p>EFO=<u>*UNCHANGED</u> / &lt;var: char: 2&gt; / &lt;c-string 2..2: c-string_2..2&gt; / *NONE</p> <p>OVERLAY=(<i>face, reverse</i>)  <i>face</i>: &lt;var: int: 1&gt; / &lt;integer 1..127&gt; / *STD / *NONE  <i>reverse</i> : &lt;var: int: 1&gt; / &lt;integer 1..127&gt; / *STD / *NONE</p> <p>FOB=<u>*UNCHANGED</u> / &lt;var: char: 4&gt; /  &lt;c-string 1..4: alphanumname_1..4, c-string_1..4&gt; / *NONE</p> <p>PAGEPCL=<u>*UNCHANGED</u> / &lt;var: int: 4&gt; / &lt;integer 1..50000&gt; / *STD</p> <p>USERRES=<u>*UNCHANGED</u> / &lt;var: char: 44&gt; /  &lt;c-string 1..44: filename_without_userid_1..44, c-string_1..44&gt; /  *STD</p>

(Teil 5 von 6)



Operation	Operanden
MODPJAT	<p> <b>TRANTAB</b>=(<i>name, file</i>)  <i>name</i>: &lt;var: char: 8&gt; /            &lt;c-string 1..8: alphanum_name_1..8, c-string_1..8&gt; / *NONE  <i>file</i>: &lt;var: char: 44&gt; /            &lt;c-string 1..44: filename_without-userid_1..44, c-string_1..44&gt; /            *SYSTEM / *STD </p> <p> <b>PRNAME</b> = <u>*UNCHANGED</u> / &lt;var: char 1..8&gt; / &lt;c-string: c-string 1..8&gt; /            *STD / *IPP </p> <p> <b>PRTYPE</b>=<u>*UNCHANGED</u> / &lt;var: enum-of _prtype_set:1&gt; / *ANY /            *LP65_PRINTER / *HP_PRINTER / *APA_PRINTER </p> <p> <b>SRTMODE</b>=<u>*UNCHANGED</u> / &lt;var: enum-of _srtmode_set:1&gt; / *NO /            *GROUP / *COLLATE / *STACKER / *AUTOMATIC </p> <p> <b>VIRTUAL</b>=<u>*UNCHANGED</u> / &lt;var: enum-of _virtual_set:1&gt; / *ALLOWED /            *NOT_ALLOWED / *STD / *MUST </p> <p> <b>PRJCLAS</b>=<u>*UNCHANGED</u> / &lt;var: int: 1&gt; / &lt;integer 1..255&gt; </p> <p> <b>PROGRAM</b>=(<i>name, string</i>)  <i>name</i>: <u>*UNCHANGED</u> / &lt;var: char: 8&gt; /            &lt;c-string 1..8: alphanum_name_1..8, c-string_1..8&gt; / *ANY  <i>string</i>: *NONE / &lt;var: char: 32&gt; /            &lt;c-string 1..32: alphanum_name_1..32, c-string_1..32&gt; </p> <p> <b>FRMNAME</b>=<u>*UNCHANGED</u> / *STD / &lt;c-string: c-string 1..63&gt; /            &lt;var: char: 63&gt; / (&lt;reg: char:63&gt;) </p> <p> <b>TOPOFF</b>=<u>*UNCHANGED</u> / *IGNORE / &lt;integer -255..255&gt; </p> <p> <b>,LEFTOFF</b>=<u>*UNCHANGED</u> / *IGNORE / &lt;integer -255..255&gt; </p> <p> <b>,SCHEDTIME</b> = (<i>schdate, schtime</i>)  <i>schdate</i>: <u>*UNCHANGED</u> / *TODAY/ &lt;var: char 1..10&gt; / &lt;c-string 10..10&gt;  <i>schtime</i>: <u>*UNCHANGED</u> / *NOW / &lt;var: char 1..5&gt; / &lt;c-string 5..5&gt; </p>

(Teil 6 von 6)

## Operandenbeschreibung

### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **CALLER=\*USER / \*SYSTEM**

Aufrufer des Makros.

### **CALLER=\*USER**

Vorgabewert für diesen Operanden. Ein Benutzer ruft das Makro auf.

### **CALLER=\*SYSTEM**

Das Makro wird vom System aufgerufen.

### **RSOPAR=\*NONE / <var: pointer>**

Adresse der RSO-Parameterliste.

### **RSOPAR=\*NONE**

Vorgabewert für diesen Operanden. Es gibt keine RSO-Parameterliste.

### **RSOPAR=<var: pointer>**

Zeiger auf die RSO-Parameterliste.

### **SPSPAR=\*NONE / <var: pointer>**

Adresse der SPS-Parameterliste.

### **SPSPAR=\*NONE**

Vorgabewert für diesen Operanden. Es gibt keine SPS-Parameterliste.

### **SPSPAR=<var: pointer>**

Zeiger auf die SPS-Parameterliste.

### **DPPAR=\*NONE / <var: pointer>**

Adresse der Dprint-Parameterliste.

### **DPPAR=\*NONE**

Vorgabewert für diesen Operanden. Es gibt keine Dprint-Parameterliste.

### **DPPAR=<var: pointer>**

Zeiger auf die Dprint-Parameterliste.

### **DSEMPAR=\*NONE / <var: pointer>**

Adresse der DSEM-Parameterliste.

### **DSEMPAR=\*NONE**

Vorgabewert für diesen Operanden. Es gibt keine DSEM-Parameterliste.

### **DSEMPAR=<var: pointer>**

Zeiger auf die DSEM-Parameterliste.

**TSN=(*tsn1*, *cluste1*)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Client-Rechner.

***tsn1*: <var: char: 4> / <c-string 4..4: alphanum-name\_4..4>**

TSN des Druckauftrags, der geändert werden soll.

***tsn1*: <var: char: 4>**

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***tsn1*: <c-string 4..4: alphanum-name\_4..4>**

Die TSN des Druckauftrags wird direkt angegeben.

***cluste1*: \*LOCAL\_CLUSTER / <var: char: 8> / <c-string 1..8: name\_1..8>**

Name des Clusters, in dem der Druckauftrag abgearbeitet werden sollte. Wenn ein Name angegeben wurde, bezieht sich die TSN auf den Gateway-Host im angegebenen Cluster. Es können nur BS2000-Cluster angegeben werden.

***cluste1*: \*LOCAL\_CLUSTER**

Vorgabewert für den Namen des Clusters, in dem der Druckauftrag abgearbeitet werden sollte.

***cluste1*: <var: char: 8>**

Der Name des Clusters wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***cluste1*: <c-string 1..8: name\_1..8>**

Der Name des Clusters wird direkt angegeben.

**SRV\_TSN=(*tsn2*, *server*)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Server-Rechner.

***tsn2*: <var: char: 4> / <c-string 4..4: alphanum-name\_4..4>**

TSN auf dem Server-Rechner des Druckauftrags, der geändert werden soll.

***tsn2*: <var: char: 4>**

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***tsn2*: <c-string 4..4: alphanum-name\_4..4>**

Die TSN des Druckauftrags wird direkt angegeben.

***server*: <var: char: 8> / <c-string 1..8: name\_1..8>**

Name des Server-Rechners, auf dem der Druckauftrag abgearbeitet wird.

***server*: <var: char: 8>**

Der Name des Server-Rechners wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***server*: <c-string 1..8: name\_1..8>**

Der Name des Server-Rechners wird direkt angegeben.

**CLT\_TSN=(*tsn2*, *server*)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Client-Rechner. Nur Druckaufträge im lokalen Cluster können auf diese Weise angesprochen werden.

Diese Option ist für den Cluster-Administrator reserviert.

*tsn2*: <var: char: 4> / <c-string 4..4: alphanum-name\_4..4>

TSN auf dem Client-Rechner des Druckauftrags, der geändert werden soll.

*tsn2*: <var: char: 4>

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*tsn2*:<c-string 4..4: alphanum-name\_4..4>

Die TSN des Druckauftrags wird direkt angegeben.

*client*: <var: char: 8> / <c-string 1..8: name\_1..8>

Name des Client-Rechners, auf dem der Druckauftrag abgearbeitet wird.

*client*: <var: char: 8>

Der Name des Client-Rechners wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*client*: <c-string 1..8: name\_1..8>

Der Name des Client-Rechners wird direkt angegeben.

**MONJV=\*NONE / <var: char: 54> / <c-string1..54: filename\_1..54\_with-catid>**

Der Druckauftrag wird über die zu diesem Auftrag gehörende MONJV angesprochen, vorausgesetzt, dass auf die angegebene MONJV auf dem lokalen Rechner zugegriffen werden kann.

**MONJV=\*NONE**

Vorgabewert für den Operanden MONJV. Es wurde keine MONJV angegeben.

**MONJV=<var: char: 54>**

Der Name der MONJV wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**MONJV=<c-string1..54: filename\_1..54\_with-catid>**

Der Name der MONJV wird direkt angegeben.

**FOREIGN=(*identif*, *cluste2*):**

Der Druckauftrag wird über eine Nicht-BS2000-Identifikation angesprochen.

*identif*: \*NONE / <var: int: 4> / <integer 1..2147483647>

Identifikation des Nicht-BS2000-Verweises auf den Druckauftrag.

*identif*: \*NONE

Vorgabewert, keine Identifikation angegeben.

*identif*: <var: int: 4>

Die Identifikation wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*identif*: <integer 1..2147483647>

Die Identifikation wird direkt angegeben.

*cluste2*: <var: char: 8> /<c-string 1..8: name\_1..8>

Name des Clusters, in dem der Druckauftrag abgearbeitet wird.

*cluste2*: <var: char: 8>

Der Name des Clusters wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*cluste2*: <c-string 1..8: name\_1..8>

Der Name des Clusters wird direkt angegeben.

**SECTREC**=(*sectid*, *sectlen*, *secttyp*, *sectpos*)

Vereinbart, ob die Datei mit Hilfe von Abschnittsmarken strukturiert werden soll. Mit "sectid" kann die Zeichenfolge (Strings) in den Datensätzen, die als Abschnittsmarken fungieren sollen, mit "sectlen" die Länge der Zeichenfolge und mit "sectpos" die Position des angegebenen Strings im Datensatz angegeben werden.

*sectid*: **\*UNCHANGED** / <var: char: 60> / <c-string 1..60: c-string\_1..60> /  
**\*WHOLE-FILE**

Es kann vereinbart werden, ob die Datei mit Hilfe von Abschnittsmarken strukturiert werden soll.

Welche Teile der in Abschnitte unterteilten Druckdatei ausgegeben werden sollen, wird mit dem Operanden FIRSREC vereinbart.

Als Abschnittsmarken können beliebige Zeichenfolgen (Strings) in den Datensätzen verwendet werden. Diese Strings können entweder in so genannten SECTION-Sätzen, die nicht gedruckt werden, stehen oder in gewöhnlichen Datensätzen der Druckdatei (die gedruckt werden). Die Zeichenfolge kann in Form abdruckbarer Zeichen oder hexadezimaler Zeichen angegeben werden.

Das Suchen nach dem Anfang des auszudruckenden Abschnitts wird in einer eigenen "Pseudo-Controller"-Task durchgeführt. Weder die Benutzer-Task noch der Drucker werden dabei gesperrt.

Ein Pseudo-Controller schreibt die Adresse des ersten Datensatzes des Abschnittes in den SPOOL-Control-Block; anschließend kann der SPOOL-OUT-Auftrag verarbeitet werden (PREPROCESSINPUT-PARTING). Wird der gewünschte Abschnitt in der Datei nicht gefunden, erscheint eine Fehlermeldung auf dem Schlussblatt (Layout des Schlussblattes siehe [Seite 611](#)).

*sectid*: **\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**sectid:** <var: char: 60>

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 60 Byte), die als String interpretiert wird, hinterlegt.

**sectid:** <c-string 1..60: c-string\_1..60>

Gibt in Hochkommata die Zeichenfolge in den Datensätzen an.

**sectid:** \*WHOLE-FILE

Die Datei soll nicht mit Hilfe von Abschnittsmarken strukturiert werden.

**sectlen:** 0 / <var: int: 1> / <integer 1..60>

Gibt die Länge der Zeichenfolge an.

**sectlen:** 0

Voreinstellung: Die Zeichenfolge hat die Länge 0 Byte.

**sectlen:** <var: int: 1>

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Länge der Zeichenfolge interpretiert wird, hinterlegt.

**ectlen:** <integer 1..60>

Die Länge der Zeichenfolge wird als Ganzzahl von 1 bis 60 angegeben.

**secttyp:** \*UNCHANGED / \*CHAR / <var: bit:1> / \*HEXA

Typ der Abschnittsidentifikation.

**secttyp:** \*UNCHANGED

Voreinstellung: Keine Änderung der Abschnittsidentifikation.

**secttyp:** \*CHAR

Die Abschnittsidentifikation besteht aus Zeichen.

**secttyp:** <var: bit:1>

Der Typ der Abschnittsidentifikation wird in einem Feld angegeben.

**secttyp:** \*HEXA

Die Abschnittsidentifikation wird sedezimal angegeben.

**sectpos:** \*UNCHANGED / <var: int: 2> / <integer 1..2047> / \*STD

Legt fest, ab welcher Position (ab welchem Byte nach dem Satzlängenfeld) der angegebene String im SECTION-Satz beginnt.

**sectpos:** \*UNCHANGED

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**sectpos:** <var: int: 2>

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Position interpretiert wird, hinterlegt.

**sectpos: <integer 1..2047>**

Ganzzahliger Wert für die Position, ab der die angegebene Zeichenfolge im SECTION-Satz beginnt.

**sectpos: \*STD**

Der gesuchte String beginnt standardmäßig am Anfang des Datensatzes:

- bei einer SAM-Datei: mit dem ersten Byte nach dem Satzlängenfeld
- bei einer ISAM-Datei mit KEY-POS=5: mit dem ersten Byte nach dem Schlüssel
- bei einer ISAM-Datei mit KEY-POS > 5: mit dem ersten Byte nach dem Satzlängenfeld.

**FIRSREC=(stpos, ststrid, ststrle, secttyp, ststrpo, ststroc)**

Vereinbart, ob von der Eingabedatei nur eine bestimmte Anzahl von Datensätzen verarbeitet werden soll. Der Anfangs- und Enddatensatz kann durch eine Satznummer, eine Abschnittsnummer oder durch eine beliebige Zeichenfolge bestimmt werden. Mit dem Operanden LASTREC können die Angaben für den Enddatensatz eingestellt werden.

Mit "stpos" kann ein Anfangsdatensatz, ab dem die Datei verarbeitet werden soll, angegeben werden.

Mit "ststrid" kann als Anfangsdatensatz ein Datensatz gewählt werden, in dem die angegebene Zeichenfolge auftritt.

Mit "ststrle" kann die Länge des Datensatzes angegeben werden.

Mit "secttyp" kann der Typ der Zeichenkette angegeben werden.

Mit "ststrpo" wird die Position innerhalb des Datensatzes, ab der der angegebene String beginnt, angegeben.

Mit "ststroc" kann vereinbart werden, ab dem wievielten Auftreten des Datensatzes, der den String enthält, die Ausgabe beginnen soll.

**stpos: <var: int: 4> / <integer 1..2147483647> / \*BEGIN\_OF\_FILE / \*BY\_STRING\_ID**

Vereinbart den Anfangsdatensatz, ab dem die Datei verarbeitet werden soll. Es kann hier der erste Datensatz der Datei, die Nummer eines Datensatzes oder einer Abschnittsmarke sowie eine beliebige Zeichenfolge in einem Datensatz gewählt werden.

**stpos: <var: int: 4>**

Der Anfangsdatensatz wird in einem Feld als Ganzzahl von 1 bis 2147483647 angegeben.

**stpos: <integer 1..2147483647>**

Der Anfangsdatensatz wird als Ganzzahl von 1 bis 2147483647 angegeben.

**stpos: \*BEGIN\_OF\_FILE**

Die Ausgabe beginnt mit dem ersten Datensatz der Datei, auch wenn SECTION-Sätze angegeben sind.

**stpos: \*BY\_STRING\_ID**

Als Anfangsdatensatz wird ein Datensatz gewählt, in dem eine nachfolgend anzugebende Zeichenfolge auftritt.

***ststrid*: \*UNCHANGED / <var: char: 60> / <c-string 1..60: c-string\_1..60>**

Vereinbart, ob die Ausgabe mit einem Datensatz beginnen soll, in dem ein bestimmter String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

***ststrid*: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

***ststrid*: <var: char: 60>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 60 Byte), die als String interpretiert wird, hinterlegt.

***ststrid*: <c-string 1..60: c-string\_1..60>**

Die Ausgabe soll mit dem Datensatz beginnen, in dem der angegebene String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

***ststrle*: 0 / <var: int: 1> / <integer 1..60>**

Gibt die Länge des Datensatzes an.

***ststrle*: 0**

Voreinstellung: Der Datensatz hat die Länge 0 Byte.

***ststrle*: <var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Länge des Datensatzes interpretiert wird, hinterlegt.

***ststrle*: <integer 1..60>**

Die Länge des Datensatzes wird als Ganzzahl von 1 bis 60 angegeben.

***sectyp*: \*UNCHANGED / \*CHAR / <var: bit:1> / \*HEXA**

Typ der Abschnittsidentifikation.

***sectyp*: \*UNCHANGED**

Voreinstellung: Keine Änderung der Abschnittsidentifikation.

***sectyp*: \*CHAR**

Die Abschnittsidentifikation besteht aus Zeichen.

***sectyp*: <var: bit:1>**

Der Typ der Abschnittsidentifikation wird in einem Feld angegeben.

***sectyp*: \*HEXA**

Die Abschnittsidentifikation wird sedezimal angegeben.

***ststrpo*: \*UNCHANGED / <var: int: 2> / <integer 1..2047> / \*STD**

Legt fest, ab welcher Position (ab welchem Byte nach dem Satzlängenfeld) der angegebene String im SECTION-Satz beginnt.

***ststrpo*: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.



***ststrpo*: <var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Position des Datensatzes interpretiert wird, hinterlegt.

***ststrpo*: <integer 1..2047>**

Angabe eines ganzzahligen Wertes für die Position des Strings im SECTION-Satz.

***ststrpo*: \*STD**

Der gesuchte String beginnt standardmäßig am Anfang des Datensatzes:

- bei einer SAM-Datei: mit dem ersten Byte nach dem Satzlängenfeld
- bei einer ISAM-Datei mit KEY-POS=5: mit dem ersten Byte nach dem Schlüssel
- bei einer ISAM-Datei mit KEY-POS > 5: mit dem ersten Byte nach dem Satzlängenfeld.

***ststroc*: \*UNCHANGED / <var: int: 2> / <integer 1..32767>**

Legt fest, ab dem wievielten Auftreten des Datensatzes, der den String enthält, die Ausgabe beginnen soll.

***ststroc*: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

***ststroc*: <var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Anzahl des Auftretens interpretiert wird, hinterlegt.

***ststroc*: <integer 1..32767>**

Ganzzahliger Wert für das Auftreten des Datensatzes.

**LASTREC=(*enpos*, *enstrid*, *enstrle*, *secttyp*, *enstrpo*, *enstroc*)**

Vereinbart den Enddatensatz, bis zu der die Datei verarbeitet werden soll. Der Enddatensatz kann durch eine Satznummer, eine Abschnittsnummer oder durch eine beliebige Zeichenfolge bestimmt werden.

Mit "enpos" kann ein Enddatensatz, bis zu dem die Datei verarbeitet werden soll, angegeben werden.

Mit "enstrid" kann als Enddatensatz ein Datensatz gewählt werden, in dem die angegebene Zeichenfolge auftritt.

Mit "enstrle" kann die Länge des Datensatzes angegeben werden.

Mit "secttyp" kann der Typ der Zeichenkette angegeben werden.

Mit "enstrpo" wird die Position innerhalb des Datensatzes, ab der der angegebene String beginnt, angegeben.

Mit "enstroc" kann vereinbart werden, ab dem wievielten Auftreten des Datensatzes, der den String enthält, die Ausgabe enden soll.

**enpos: <var: int: 4> / <integer 1..2147483647> / \*END\_OF\_FILE / \*BY\_STRING\_ID**  
Vereinbart den Enddatensatz, bis zu dem die Datei verarbeitet werden soll. Es kann hier der letzte Datensatz der Datei, die Nummer eines Datensatzes oder einer Abschnittsmarke sowie eine beliebige Zeichenfolge in einem Datensatz gewählt werden.

**enpos: <var: int: 4>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 4 Byte), die als Nummer des Enddatensatzes interpretiert wird, hinterlegt.

**enpos: <integer 1..2147483647>**

Der Enddatensatz wird als Ganzzahl von 1 bis 2147483647 angegeben.

**enpos: \*END\_OF\_FILE**

Die Ausgabe endet mit dem letzten Datensatz der Datei, auch wenn SECTION-Sätze angegeben sind.

**enpos: \*BY\_STRING\_ID**

Als Enddatensatz wird ein Datensatz gewählt, in dem eine nachfolgend anzugebende Zeichenfolge auftritt.

**enstrid: \*UNCHANGED / <var: char: 60> / <c-string 1..60: c-string\_1..60>**

Vereinbart, ob die Ausgabe mit einem Datensatz enden soll, in dem ein bestimmter String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

**enstrid: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**enstrid: <var: char: 60>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 60 Byte), die als String interpretiert wird, hinterlegt.

**enstrid: <c-string 1..60: c-string\_1..60>**

Die Ausgabe soll mit dem Datensatz enden, in dem der angegebene String abdruckbarer oder hexadezimaler Zeichen an einer bestimmten Stelle in der Datei aufgefunden wird.

**enstrle: 0 / <var: int: 1> / <integer 1..60>**

Gibt die Länge des Datensatzes an.

**enstrle: 0**

Voreinstellung: Der Datensatz hat die Länge 0 Byte.

**enstrle: <var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Länge des Datensatzes interpretiert wird, hinterlegt.

**enstrle: <integer 1..60>**

Die Länge des Datensatzes wird als Ganzzahl von 1 bis 60 angegeben.

**sectyp: \*UNCHANGED / \*CHAR <var: bit:1> / \*HEXA**

Typ der Abschnittsidentifikation.

**sectyp: \*UNCHANGED**

Voreinstellung: Keine Änderung der Abschnittsidentifikation.

**sectyp: \*CHAR**

Die Abschnittsidentifikation besteht aus Zeichen.

**sectyp: <var: bit:1>**

Der Typ der Abschnittsidentifikation wird in einem Feld angegeben.

**sectyp: \*HEXA**

Die Abschnittsidentifikation wird sedezimal angegeben.

**enstrpo: \*UNCHANGED / <var: int: 2> / <integer 1..2047> / \*STD**

Legt fest, ab welcher Position (ab welchem Byte nach dem Satzlängengebiet) der angegebene String im SECTION-Satz beginnt.

**enstrpo: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**enstrpo: <var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Position des Datensatzes interpretiert wird, hinterlegt.

**enstrpo: <integer 1..2047>**

Angabe eines ganzzahligen Wertes für die Position des Strings im SECTION-Satz.

**enstrpo: \*STD**

Der gesuchte String beginnt standardmäßig am Anfang des Datensatzes:

- bei einer SAM-Datei: mit dem ersten Byte nach dem Satzlängengebiet
- bei einer ISAM-Datei mit KEY-POS=5: mit dem ersten Byte nach dem Schlüssel
- bei einer ISAM-Datei mit KEY-POS > 5: mit dem ersten Byte nach dem Satzlängengebiet.

**enstroc: \*UNCHANGED / <var: int: 2> / <integer 1..32767>**

Legt fest, ab dem wievielten Auftreten des Datensatzes, der den LASTREC-String enthält, die Ausgabe abschließen soll.

**enstroc: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**enstroc: <var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Anzahl des Auftretens interpretiert wird, hinterlegt.

**enstroc: <integer 1..32767>**

Ganzzahliger Wert für das Auftreten des Datensatzes.

**RECPART=(*firstch*, *lastch*)**

Vereinbart, dass von allen ausgewählten Datensätzen nur ein bestimmter Teil jedes Satzes verarbeitet werden soll.

***firstch*: \*UNCHANGED / <var: int: 2> / <integer 1..32767>**

Ermöglicht die Angabe einer Byte-Nummer (Satzspalte), ab der die Datensätze einer Datei auszugeben sind. (Die Bytes eines Datensatzes sind von links nach rechts - mit 1 beginnend - durchnummeriert; ISAM-Schlüssel und Steuerzeichen sind Bestandteile eines Datensatzes).

Es ist zu unterscheiden, ob im Operanden LINESP einer der Werte \*BY\_EBCDIC\_CONTROL , \*BY\_IBM\_CONTROL oder \*BY\_ASA\_CONTROL angegeben ist oder nicht.

*Ist einer dieser Werte angegeben:*

Die Ausgabe beginnt mit dem der angegebenen Byte-Nummer folgenden Datenbyte. Das Vorschubsteuerzeichen wird unabhängig von der Angabe bei *first* ausgewertet, solange der Wert für *first* kleiner ist als die Länge des Datensatzes.

Ist der angegebene Wert > der Länge des Datensatzes, wird dieser ignoriert (d.h. es wird nicht gedruckt und nicht vorgeschoben).

*Ist keiner dieser Werte angegeben:*

Die Ausgabe beginnt mit dem der angegebenen Byte-Nummer entsprechenden Datenbyte.

***firstch*: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

***firstch*: <var: int: 2>**

Name eines Feldes, das mit FL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Ganzzahl (Länge 2 Byte), die als Satzspalte interpretiert wird, hinterlegt.

***firstch*: <integer 1..32767>**

Ganzzahliger Wert für die Byte-Nummer (Satzspalte), ab der die Datensätze einer Datei auszugeben sind.

***lastch*: \*UNCHANGED / <var: int: 2> / <integer 1..32767>**

Diese Angabe bezeichnet das letzte Byte, das von jedem Datensatz noch gedruckt werden soll.

Es ist zu unterscheiden, ob im Operanden LINESP einer der Werte \*BY\_EBCDIC\_CONTROL , \*BY\_IBM\_CONTROL oder \*BY\_ASA\_CONTROL angegeben ist oder nicht.

*Ist einer dieser Werte angegeben:*

Die Ausgabe endet mit dem der angegebenen Byte-Nummer folgenden Datenbyte (Ausnahme: Bei der Angabe FIRST-CHARACTER für eine ISAM-Datei mit KEY-POSITION=5 endet die Ausgabe mit dem der angegebenen Byte-Nummer entsprechenden Datenbyte).

*Ist keiner dieser Werte angegeben:*

Die Ausgabe endet mit dem der angegebenen Byte-Nummer entsprechenden Datenbyte.

Sind die Datensätze länger als nach der Formulardefinition erlaubt, werden sie in der Folgezeile weitergedruckt.

***lastch***: **\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

***lastch***: **<var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als letztes Byte interpretiert wird, hinterlegt.

***lastch***: **<integer 1..32767>**

Ganzzahliger Wert für das letzte Byte, das von jedem Datensatz noch gedruckt werden soll.

**OUTPART**=(*from, to, dim*)

Vereinbart, dass die gesamte Eingabedatei druckaufbereitet werden, die Ausgabe sich jedoch auf eine Untermenge aller logischen Druckseiten beschränken soll.

***from***: **\*UNCHANGED** / **<var: int: 4>** / **<integer -2147483647..2147483647>** / **\*BEGIN\_OF\_FILE**

Ermöglicht die Angabe einer Seiten- oder Zeilennummer der Druckdatei, mit der die Ausgabe beginnen soll, was durch den Operanden *dimension* spezifiziert wird.

***from***: **\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

***from***: **<var: int: 4>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 4 Byte), die als Seiten- oder Zeilennummer interpretiert wird, hinterlegt.

***from***: **<integer -2147483647..2147483647>**

Ganzzahliger Wert für eine Seiten- oder Zeilennummer der Druckdatei. Bei einem Wert < 0 wird die Anzahl auszugebender Seiten oder Zeilen vom Dateiende her berechnet. Bei dem Operanden DIMENSION wird festgelegt, ob die Angabe als Seiten- oder Zeilennummer interpretiert werden soll.

***from***: **\*BEGIN\_OF\_FILE**

Die Ausgabe beginnt mit dem Dateianfang. Die Seiten einer Druckdatei werden wie beim Operanden LINEPP beschrieben festgelegt (soweit nicht durch Steuerzeichen ein vorzeitiger Seitenvorschub erfolgt).

***to***: **\*UNCHANGED** / **<var: int: 4>** / **<integer 1..2147483647>** / **\*END\_OF\_FILE**

Ermöglicht die Angabe einer Seiten- oder Zeilennummer der Druckdatei, mit der die Ausgabe beendet werden soll, was durch den Operanden *dim* spezifiziert wird.

Der hier angegebene Wert muss größer sein als der bei dem Operanden *from* angegebene.

**to: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**to: <var: int: 4>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 4 Byte), die als Seiten- oder Zeilennummer interpretiert wird, hinterlegt.

**to: <integer 1..2147483647>**

Ganzzahliger Wert für eine Seiten- oder Zeilennummer der Druckdatei.

**to: \*END\_OF\_FILE**

Die Ausgabe endet mit dem Dateiende. Die Seiten einer Druckdatei werden wie beim Operanden LINEPP beschrieben festgelegt (soweit nicht durch Steuerzeichen ein vorzeitiger Seitenvorschub erfolgt).

**dim: \*UNCHANGED / <var: enum-of \_layout\_part\_dim\_set: 1> / \*PAGES / \*LINES**

Legt fest, ob die bei den Operanden *from* und *to* angegebenen Werte als Seiten- oder Zeilennummern zu interpretieren sind.

**dim: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**dim: <var: enum-of \_layout\_part\_dim\_set: 1>**

Die Vereinbarung bzgl. der Dimension wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden.

**dim: \*PAGES**

Die bei den Operanden *from* und *to* angegebenen Werte sind als Seitennummern zu interpretieren.

**dim: \*LINES**

Die bei den Operanden *from* und *to* angegebenen Werte sind als Zeilennummern zu interpretieren.

**DOCFORM=\*UNCHANGED / <var: enum-of \_docform\_set: 1> / \*TEXT / \*PAGE\_FORMAT / \*SPECIAL\_FORMAT**

Legt fest, welcher Art die Dokumentinhalte sind, d.h. welches Format die auszudruckende Datei bezüglich der Auswertung von Vorschubsteuerzeichen, Druckersteuerzeichen, Fontkennzeichen, RENO-Kommandos und/oder VTSU-Codes aufweist.

**DOCFORM=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**DOCFORM=<var: enum-of \_docform\_set: 1>**

Die Vereinbarung bzgl. der Art des Dokumentinhaltes wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*TEXT
1	*PAGE_FORMAT
2	*SPECIAL_FORMAT
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**DOCFORM=\*TEXT**

Die Datei weist bis auf Vorschubsteuerzeichen keine druckerspezifischen Steuerzeichen auf. Im Operand LINESP kann die Position des Vorschubsteuerzeichens im Datensatz angegeben werden.

Die Daten werden ohne irgendeine Veränderung an den Drucker geschickt. Da nur die auszudruckenden Daten an den Drucker geschickt werden, können Datensätze „abgeschnitten“ werden.

Für HP- und HP90-Drucker gilt:

Bei Angabe des Operanden PRATYPE=\*ANY wird das Zeichen X'FF' durch das Zeichen X'1F' ersetzt, um kompatibel zu sein mit der PRM-Anweisung CONVERT-PRINT-RESOURCES.

Für LP65-Drucker gilt:

SPOOLOUT-Aufträge, bei denen DOCFORM=\*TEXT spezifiziert ist, können beliebige LP65-Steuerzeichen und Druckersteuerzeichen enthalten. Nur Sätze, die länger als 8192 Zeichen sind, werden abgeschnitten.

Für RSO gilt:

Datensätze der Typen A-1 oder A-2 sollen ausgedruckt werden (keine Steuerzeichen im Datenstrom), d.h. außer Vorschubsteuerzeichen in der ersten Spalte der Datensätze werden keine Steuerzeichen ausgewertet. Aus diesem Grunde ist auch kein Fontwechsel möglich, d.h. es wird mit dem Standardfont des Formulars oder, falls angegeben, mit dem ersten bei CHARSET angegebenen Font ausgedruckt. Nichtabdruckbare Zeichen, d.h. Zeichen mit einem sedezimalen Wert < X'40', werden als Leerzeichen ausgegeben.

Übersteigt die Satzlänge die max. Zeilenlänge, so wird der Datensatz beim Ausdruck abgeschnitten. Die max. Zeilenlänge ist abhängig vom Zeichenabstand; dieser ist durch den verwendeten Font (siehe Operand CHARSET) definiert.

**DOCFORM=\*PAGE\_FORMAT**

Die Datei weist spezifische Steuerzeichen für Laserdrucker auf. Mit dem Operanden CON-TMOD kann festgelegt werden, wie die Steuerzeichen interpretiert werden sollen.

**DOCFORM=\*SPECIAL\_FORMAT**

Im Dokument wird eine druckerspezifische Sprache verwendet. In diesem Fall stellen die Subsysteme SPOOL und Distributed Print Services (Dprint) eine transparente Steuerung für das Dokument zur Verfügung, d.h. die Datei wird ohne Auswertung an einen RSO- oder Xprint-Drucker übertragen.

**LINESP=(spacing, ccpos)**

Bestimmt die Anzahl der Zeilenvorschübe bzw. die Art der Steuerzeichenauswertung.

*spacing:* **\*UNCHANGED** / <var: enum-of \_space\_set:1> / **\*SPACE\_1** / **\*SPACE\_2** / **\*SPACE\_3** / **\*BY\_ASA\_CONTROL** / **\*BY\_EBCDIC\_CONTROL** / **\*BY\_IBM\_CONTROL** / **\*NO**

Entsprechend des Operandenwertes kann Folgendes vereinbart werden:

Die Datensätze sollen mit 1 / 2 / 3 Zeilen Abstand (**\*SPACE\_n**) ausgedruckt werden.

Der Inhalt im ersten Byte der Datensätze soll als ASA-Vorschubsteuerzeichen interpretiert werden (**\*BY\_ASA\_CONTROL**).

Der Inhalt im ersten Byte der Datensätze soll als EBCDIC-Vorschubsteuerzeichen interpretiert werden (**\*BY\_EBCDIC\_CONTROL**).

Der Inhalt im ersten Byte der Datensätze soll als IBM-Vorschubsteuerzeichen interpretiert werden (**\*BY\_IBM\_CONTROL**).

Der Inhalt im ersten Byte der Datensätze soll nicht als Vorschubsteuerzeichen interpretiert werden (**\*NO**).

Der Standardwert (**\*STD**) hängt vom Modus der Steuerzeichenauswertung (Operand CONTMOD) ab.

*spacing:* **\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

*spacing:* <var: enum-of \_space\_set:1>

Die Vereinbarung bzgl. der Anzahl der Zeilenvorschübe wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*STD
1	*SPACE_1
2	*SPACE_2
4	*SPACE_3
8	*BY_EBCDIC_CONTROL
16	*BY_ASA_CONTROL
32	*BY_IBM_CONTROL
144	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	



**ccpos: \*UNCHANGED / <var: int: 2> / <integer 1..2040> / \*STD**

Nummer des Datenbytes, in dem SPOOL das Vorschubsteuerzeichen findet. Bei Datensätzen variabler Länge werden die Felder, in denen die Länge steht, nicht zu den Daten gerechnet, d.h. nicht mitgezählt.

**ccpos: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**ccpos: <var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Nummer des Datenbytes interpretiert wird, hinterlegt.

**ccpos: <integer 1..2040>**

Ganzzahliger Wert für die Nummer des Datenbytes.

**ccpos: \*STD**

Der Standardwert hängt vom Modus der Steuerzeichenauswertung (Operand CONTMOD) ab.

**CONTMOD=(mode, pcc, ctltype)**

Vereinbart, wie die Steuerzeichen interpretiert werden sollen (*mode*), ob die Steuerzeichenleiste am Anfang jeder Seite vorhanden sein muss (*pcc*) und ob die Steuerzeichen zur Verarbeitung auf Druckern des Typs HP bzw. HP90 geeignet (*type*) sind.

**mode: \*UNCHANGED / <var: enum-of \_control\_mode\_set: 1> / \*STD / \*PHYSICAL / \*LINE\_MODE / \*LOGICAL / \*APA / \*PAGE\_MODE**

Vereinbart, wie die Steuerzeichen interpretiert werden sollen.

**mode: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**mode: <var: enum-of \_control\_mode\_set: 1>**

Die Vereinbarung, wie die Steuerzeichen interpretiert werden sollen, wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*STD
2	*PHYSICAL
4	*PAGE_MODE
8	*LOGICAL
16	*LINE_MODE
64	*APA
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**mode: \*STD**

Der Operand soll nicht ausgewertet werden.

**mode: \*PHYSICAL**

Nur für RSO.

Es wird automatisch LINESP=\*NO gesetzt, d.h. Sie müssen Seiten- und Zeilenvorschub mit LINE-MODE-Steuerzeichen (d.h. VTSU-Codes, Druckersteuerzeichen, RENO-Kommandos) in der Datei selbst realisieren.

Die Angabe LINESP=\*BY\_EBCDIC\_CONTROL bei Datensätzen vom Typ D-2 bewirkt, dass das Vorschubsteuerzeichen im ersten Byte als Zeilen- oder Seitenvorschubsteuerzeichen ausgewertet wird. Es werden auch nichtabdruckbare Zeichen zum Drucker übertragen (im Unterschied zu CONTMOD=\*LINE\_MODE). Sie sind selbst verantwortlich für die Richtigkeit der Steuerzeichen in der Datei (einschließlich der Vorschubsteuerung).

**mode: \*LINE\_MODE**

Nur für RSO.

Es werden Datensätze vom Typ C (siehe Anhang) ausgedruckt. Die Datensätze können die Daten gemischt mit LINE-MODE-Steuerzeichen enthalten. LINE-MODE-Steuerzeichen sind (in beliebiger Kombination)

- Druckersteuerzeichen (d.h. physikalische Steuerzeichen beginnend mit X'27' oder X'3C')
- RENO-Kommandos
- VTSU-Codes.

Die Kontrolle über Satz- und Dateiaufbau (einschließlich Seiten- und Zeilenvorschub über LINE-MODE-Steuerzeichen) liegt allein beim Anwender. Er muss beim Seitendrucker 9025/9026-RENO auch für die richtige Anfangspositionierung auf dem Papier sorgen.

Ein Loop wird nicht ausgewertet. Nichtabdruckbare Zeichen, d.h. Zeichen mit einem sedezimalen Wert < X'40' werden als Leerzeichen ausgegeben.

Der SPOOLOUT-Auftrag wird mit dem Standardfont des verwendeten Formulars ausgeführt, bis Sie den Font innerhalb der Datei mittels LINE-MODE-Steuerzeichen wechseln.

**mode: \*LOGICAL**

Nur für RSO-Drucker.

Gibt an, dass Datensätze vom Typ B-1 oder B-2 auszudrucken sind, d.h. Datensätze, die außer einem Vorschubsteuerzeichen im ersten Byte auch Daten gemischt mit Fontkennzeichen, Druckersteuerzeichen, RENO-Kommandos und VTSU-Codes enthalten können (siehe Handbuch „[RSO \(BS2000/OSD\)](#)“, Abschnitt Datensatztyp Typ B-1/B-2). Mit Ausnahme der VTSU-Codes VPA, NP, VT, NL und CR, die als Leerzeichen ausgegeben werden, werden die oben genannten Steuerzeichen ausgewertet.

Ein Fontkennzeichen, ein VTSU-Code oder ein RENO-Kommando bleibt gültig, bis ein neues Steuerzeichen angegeben wird.

Bei fehlenden Fontkennzeichen wird mit dem Standardfont des Formulars gedruckt.

Da der Seitenvorschub bei `CONTMOD=*LOGICAL` über einen Loop oder konstanten Zeilenvorschub realisiert wird, sollten auch die `RENO`-Kommandos `\LF`, `\FF` und `\CR` nicht in der Datei vorkommen. Auch das Setzen der Formularhöhe ist nicht erlaubt.

**mode: \*APA**

Gibt an, dass die auszudruckende Datei APA-Druckersteuerzeichen enthält, die ausgewertet werden sollen.

**mode: \*PAGE\_MODE**

Die Steuerzeichen sollen als spezifische Steuerzeichen für Seitendrucker interpretiert werden.

**pcc: \*UNCHANGED / <var enum-of \_pcc\_set:1> / \*YES / \*NO**

Legt fest, ob die Steuerzeichenleiste am Anfang jeder Seite (d.h. im Loop immer nach Sprung auf den Vertikaltabulator "Kanal 1" ) vorhanden sein muss. Der Operand wird nur bei Ausgabe auf HP und HP90-Druckern ausgewertet (Einstellung `CONTMODE=*PAGE_MODE`).

**pcc: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**pcc: <var enum-of \_pcc\_set:1>**

Die Vereinbarung bzgl. der Steuerzeichenleiste wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (`EQUATE`) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*YES
1	*NO
2	*DUMMY
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**pcc: \*YES**

Die Steuerzeichenleiste muss vorhanden sein.

**pcc: \*NO**

Keine Steuerzeichenleiste am Seitenanfang. Dies bewirkt jedoch, dass bei Ausgabe auf HP-Drucker folgende Funktionen nicht gesteuert werden können:

- Film-Dia auf einzelnen Seiten der Datei; im Makro MODPJAT angegeben, wird ein Film-Dia auf jeder Seite der Druckdatei des SPOOLOUT-Auftrags verwendet.
- Seitenkopien für einzelne Seiten der Datei; alle Seiten der Druckdatei werden mit so vielen Kopien ausgegeben, wie im Makro MODPJAT angegeben wurde.
- Spaltenweises Einrücken auf einzelnen Seiten; der im Makro MODPJAT angegebene Wert gilt für alle Seiten der Druckdatei.

- FOB-Datendia auf einzelnen Seiten der Datei; im Makro MODPJAT angegeben, wird ein FOB-Datendia auf jeder Seite der Druckdatei des SPOOLOUT-Auftrags verwendet.
- Die Angabe einer Kopienreferenznummer ist nicht möglich.
- Steuerung der Seitendrehung für einzelne Seiten der Druckdatei; alle Seiten werden entweder im Hochformat oder im Querformat ausgegeben, so wie dies im Makro MODPJAT angegeben ist.

*ctltype:* **\*UNCHANGED** / <var: enum-of \_control\_type\_set:1> / **\*COMPATIBLE** / **\*HP**  
 Legt fest, ob die Steuerzeichen zur Verarbeitung auf Druckern des Typs HP bzw. HP90 geeignet sind oder in diese Form konvertiert werden müssen.

*ctltype:* **\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

*ctltype:* <var: enum-of \_control\_type\_set:1>

Die Vereinbarung bzgl. der Konvertierung der Steuerzeichen wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*DUMMY
1	*COMPATIBLE
2	*HP
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

*ctltype:* **\*COMPATIBLE**

In der Datei sind keine für Drucker des Typs HP bzw. HP90 spezifischen Steuerzeichen hinterlegt; die Steuerzeichen müssen von SPOOL in diese Form konvertiert werden.

*ctltype:* **\*HP**

In der Datei sind spezifische Steuerzeichen für einen Drucker des Typs HP bzw. HP90 hinterlegt, die auch nur von Druckern dieses Typs verarbeitet werden können.

**ADDCOP=\*UNCHANGED** / <var: int: 1> / <integer 0..255>

Legt fest, wie oft die Datei zusätzlich auszudrucken ist. Jeder zusätzliche Ausdruck erhält ein eigenes Deckblatt. Als Wert dieses Operanden kann eine Ganzzahl von 0 bis 255 oder der Name eines Feldes, das mit FL definiert ist (Länge 1 Byte) bzw. ein Register, in dem der Wert steht, angegeben werden.

Voreinstellung: 0; kein zusätzlicher Ausdruck.

**ADDCOP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**ADDCOP=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), hinterlegt, die als Anzahl der zusätzlichen Ausdrücke interpretiert wird.

**ADDCOP=<integer 0..255>**

Anzahl der zusätzlichen Ausdrücke.

**CHECKP=\*UNCHANGED / <var: enum-of \_checkp\_set:1> / \*ON\_PAGES / \*ON\_SECTION\_RECORDS**

Legt fest, ob die Checkpoint-Verarbeitung durch den Controller auf der Basis von Seiten oder von Abschnitten (SECTIONS) durchgeführt werden soll.

**CHECKP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**CHECKP=<var: enum-of \_checkp\_set:1>**

Die Vereinbarung bzgl. des Wiederanlaufs wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*ON_PAGES
1	*ON_SECTION_RECORDS
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**CHECKP=\*ON\_PAGES**

Standard-Wiederaufsetzverfahren.

Die Verarbeitung wird - bei Wiederanlauf eines unterbrochenen Jobs - eine gegebene Anzahl von Seiten zurückversetzt wieder aufgenommen.

**CHECKP=\*ON\_SECTION\_RECORDS**

Der Operandenwert kann für alle Druckertypen angegeben werden, bietet aber vor allem bei Ausgabe auf Drucker der Typen HP90 im TWO-UP-Verfahren und LP65 Vorteile:

Eine physische Seite kann bei diesem Druckertyp mehrere logische Seiten umfassen, ohne dass SPOOL dies feststellen kann (Information steht in der PCL-Datei), d.h. das an logischen Seiten orientierte Standard-Wiederaufsetzverfahren ist bei diesem Druckertyp sehr fehleranfällig.

SECTION-Sätze dienen hier als Markierungen zum Wiederaufsetzen. Sie teilen mit Hilfe von SECTION-Sätzen Ihre Datei in verschiedene Abschnitte ein. Diese SECTION-Sätze müssen die zu einer korrekten Verarbeitung der Daten notwendigen Druckerbefehle enthalten. Sind in einer physischen Seite mehrere logische Seiten enthalten, muss im SECTION-Satz zudem der Beginn einer physischen Seite deutlich gekennzeichnet sein.

Nach Auftreten eines Fehlers wird bei HOLD-PRINT-JOB und RESUME-PRINT-JOB die Verarbeitung eine gegebene Anzahl von Abschnitten (SECTIONS) weiter vorne in der Datei wieder aufgenommen; d.h. die Angabe bei RESTART-POSITION=\*PAGE(...) bzw. \*BACK(...) bezeichnet hier nicht eine bestimmte Anzahl von Seiten, sondern von Abschnitten (SECTIONS). Auch die in Fehlermeldungen ausgegebenen Zahlen sind keine Seitenangaben, sondern Abschnittsangaben.

Um zu einem korrekten Ergebnis zu kommen, muss ein Abschnitt mindestens mit einer physischen Seite korrespondieren, im Idealfall genau einer Seite. Wird zusammen mit CHECKP=\*ON\_SECTION\_RECORDS der Operandenwert CONTMOD=\*PAGE\_MODE angegeben, haben Sie für Ausgabe auf Laserdrucker sicherzustellen, dass die SECTION-Sätze unmittelbar vor Datensätzen stehen, die die Steuerzeichenleiste am Anfang der Druckseite enthalten. Sobald neu positioniert wird (PRNTDOC oder Wiederanlauf eines unterbrochenen Jobs), muss der erste Datensatz, den SPOOL druckt, immer die Steuerzeichenleiste enthalten.

**PRJNAME=\*UNCHANGED / <var: char: 8> /**

**<c-string 1..8: c-string\_with-lower\_case\_1..8>**

Auftragsname für den SPOOLOUT-Auftrag.

Der Auftragsname kann aus maximal 8 Zeichen aus der Menge (A,...Z,0,...9,@, #,\$,..) gebildet werden, darf aber nicht mit einem Bindestrich beginnen oder mit einem Punkt enden. Er darf nur dann mit einem Punkt beginnen, wenn diesem ein alphabetisches Zeichen folgt; der Punkt selbst als Bestandteil des Auftragsnamens wird in diesem Fall nicht auf dem Deckblatt abgedruckt. Die Sonderzeichenfolge Punkt und Bindestrich '.'- darf nur in Hochkommata angegeben werden.

Der Auftragsname wird auf das Deckblatt in der dritten Großdruckzeile gedruckt und erscheint auch in der Ausgabe nach dem Kommando SHOW-PRINT-JOB-STATUS. Bei fehlender Angabe wird an seiner Stelle der Auftragsname aus dem Kommando SET-LOGON-PARAMETERS eingesetzt.

**PRJNAME=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PRJNAME=<var: char: 8>**

Name eines Feldes, das mit CL definiert. In diesem Feld wird eine Zeichenfolge (Länge 8 Byte), die als Auftragsname interpretiert wird, hinterlegt.

**PRJNAME=<c-string 1..8: c-string\_with-lower\_case\_1..8>**

Auftragsname, der für den SPOOLOUT-Auftrag vergeben werden soll (und einen zusätzlich vergebenen Auftragsnamen für den kommandogebenden Auftrag überschreibt).

**PRJPRIO=\*UNCHANGED / <var: int: 1> / <integer 30..255>**

Bestimmt die Dringlichkeit des SPOOLOUT-Auftrags, mit der dieser relativ zu anderen SPOOLOUT-Aufträgen gestartet wird.

**PRJPRIO=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PRJPRIO=<var: int: 1>**

Name eines Feldes, das mit FL definiert. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Priorität interpretiert wird, hinterlegt.

**PRJPRIO=<integer 30..255>**

Ganzzahliger Wert für die Priorität, die der SPOOLOUT-Auftrag erhalten soll.

Die höchste Priorität, die Sie hier vergeben können, ist im Benutzerkatalog festgelegt und kann mit dem Kommando SHOW-USER-ATTRIBUTES abgefragt werden. Bei unzulässiger Angabe wird (wie bei fehlender Angabe) für den SPOOLOUT-Auftrag die gleiche Priorität vergeben, die der kommandogebende Auftrag auch hat.

**TRUNC=\*UNCHANGED / <var: enum-of \_truncation\_set:1> / \*STD / \*DELETE\_FILE / \*KEEP\_FILE**

Legt das Verhalten für den Fall fest, dass Zeilen abgeschnitten werden.

**TRUNC=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**TRUNC=<var: enum-of \_truncation\_set:1>**

Die Vereinbarung bzgl. der Fortsetzung des SPOOLOUT-Auftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*STD
1	*DELETE_FILE
2	*KEEP_FILE
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**TRUNC=\*STD**

Es soll die Voreinstellung aus der SPOOL-Parameterdatei gültig sein. Sie können sich diesen Wert mit dem Kommando bzw. der SPSEIVE-Anweisung SHOW-SPOOL-PARAMETERS ausgeben lassen (Feld: ERROR-PR=(TRUNC=)).

**TRUNC=\*DELETE\_FILE**

Die Verarbeitung des SPOOLOUT-Auftrags wird fortgesetzt (d.h. DELF=\*YES wird ausgeführt).

Auf dem Schlussblatt wird eine entsprechende Warnung ausgedruckt.

**TRUNC=\*KEEP\_FILE**

Die Verarbeitung des SPOOLOUT-Auftrags wird fortgesetzt, jedoch die Datei anschließend nicht gelöscht.

**LOCKF=\*UNCHANGED / <var: enum-of \_lockf\_set:1> / \*YES / \*NO**

Legt fest, ob die Datei geschützt werden soll, solange sich der SPOOL-OUT-Auftrag im Wartezustand befindet (TYPE 4, siehe Ausgabe des Kommandos SHOW-PRINT-JOB-STATUS). Die Datei kann in dieser Zeit nur gelesen werden.

Banddateien werden prinzipiell nicht gesperrt. Während der Verarbeitung des SPOOL-OUT-Auftrages (TYPE 5, siehe Ausgabe des Kommandos SHOW-PRINT-JOB-STATUS) ist die Datei prinzipiell geschützt - unabhängig von der Angabe bei diesem Operanden.

Ein SPOOL-OUT-Auftrag wird auch dann erzeugt, wenn die auszugebende Datei durch das Kommando SECURE-RESOURCE-ALLOCATION reserviert ist. Zum Zeitpunkt der Verarbeitung des SPOOL-OUT-Auftrags muss die Reservierung jedoch aufgehoben sein – sonst wird der Auftrag nicht ausgeführt.

Die auszugebende Datei bleibt bis zum Ende der Session gesperrt, wenn im Makro MODPJAT der Operand LOCKF=\*YES angegeben wird und der Auftrag wegen der Reservierung nicht ausgeführt werden kann.

**LOCKF=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**LOCKF=<var: enum-of \_lockf\_set:1>**

Die Vereinbarung bzgl. des Dateischutzes wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

**LOCKF=\*YES**

Die Datei ist geschützt, während der SPOOL-OUT-Auftrag sich im Wartezustand befindet.

LOCKF=\*YES wird ignoriert, wenn zugleich im Operanden DTYPE einer der Werte \*EAM oder \*SYSTEM\_FILE angegeben ist.

Ein Makro MODPJAT auf ein Bibliothekselement mit gleichzeitiger Angabe von LOCKF=\*YES wird abgewiesen.

Der Dateischutz durch LOCKF=\*YES bleibt auch bestehen, wenn der SPOOL-OUT-Auftrag erst im nächsten Systemlauf verarbeitet wird.

**LOCKF=\*NO**

Die Datei ist nicht geschützt, während der SPOOL-OUT-Auftrag sich im Wartezustand befindet. Die Datei kann bis zum Beginn der Verarbeitung des SPOOL-OUT-Auftrags gelöscht oder geändert werden.

LOCKF=\*NO wird ignoriert für temporäre Dateien.



**DELFL=\***UNCHANGED / <var: enum-of \_delf\_set: 1> / \*NO / \*YES / \*DESTROY

Legt fest, ob die Datei nach Beendigung der Ausgabe gelöscht werden soll und wenn ja, ob zusätzlich ihr Katalogeintrag und die Daten mit X'00..0' überschrieben werden sollen. Voreinstellung: Die Datei wird nach dem Drucken nicht gelöscht und nicht mit binären Nullen überschrieben. Sie müssen zum Schreibzugriff auf die Datei berechtigt sein. Gehört die auszugebende Datei zu einer Dateigenerationsgruppe, wird der Operand DELF ignoriert.

Sollen in einem SPOOLOUT-Auftrag mehrere Elemente einer PLAM-Bibliothek mit SECTREC=... ausgegeben werden, wird der Operand DELF auf \*NO gesetzt (d.h. unterdrückt). Der Operand darf nicht zusammen mit \*SYSLST, \*SYSLSTnn oder \*SYSOUT angegeben werden.

**DELFL=\***UNCHANGED

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**DELFL=<var: enum-of \_delf\_set: 1>**

Die Vereinbarung bzgl. des Löschens der Dateien wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*YES
2	*DESTROY
3	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**DELFL=\***NO

Die Datei soll nach dem Drucken nicht gelöscht werden (Ausnahme: EAM- und Systemdateien).

**DELFL=\***YES

Die Datei soll gelöscht werden, sobald die Ausgabe beendet ist.

**DELFL=\***DESTROY

Nicht für EAM- und katalogisierte Systemdateien.

Legt fest, dass nach dem Drucken der Datei sowohl ihr Katalogeintrag als auch die Daten selbst mit binär Null überschrieben werden sollen.

**PAGECOP=\*UNCHANGED / <var: int: 2> / <integer 0..255> / \*STD**

Legt fest, wie oft jede einzelne Seite aufeinander folgend zu wiederholen ist. Zusätzliche Ausdrücke der ganzen Datei können mit dem Operanden ADDCOP angefordert werden. Jeder Makro MODPJAT mit dem Operanden PAGECOP wird zurückgewiesen, wenn bei gleichzeitiger Angabe des Operanden LINEPP die darin angegebene Zeilenzahl größer ist als die Anzahl der Zeilen im Loop, vermindert um die Anzahl der Zeilen vor der Zeile, auf der Vertikaltabulator "Kanal 1" definiert ist.

Auf einem HP- bzw. HP90-Drucker oder einem 2050- bzw. 2090-APA-Drucker (auch Zwilingsdrucker) können maximal 255 Exemplare einer Seite hintereinander gedruckt werden. PAGECOP=255 hat die gleiche Wirkung wie PAGECOP=254: Ein Original und 254 Kopien werden gedruckt.

Im lokalen SPOOL-Betrieb nur für die Druckertypen HP, HP90, LP65.

**PAGECOP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PAGECOP=<var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Anzahl der Seitenkopien interpretiert wird, hinterlegt.

**PAGECOP=<integer 0..255>**

Ganzzahliger Wert für die Anzahl der Seitenkopien.

**PAGECOP=\*STD**

Für LP65-Drucker:

Anzahl der Seitenkopien wie in der PCL-Datei angegeben.

Für alle übrigen Druckertypen:

PAGECOP=0.

**LEFTMAR=\*UNCHANGED / <var: int: 1> / <integer 0..31> / \*STD**

Vereinbart, ob der Ausgabetext eingerückt werden soll.

Der Operand LEFTMAR wird ignoriert, wenn zugleich CONTMOD=\*PHYSICAL angegeben ist.

**LEFTMAR=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**LEFTMAR=<var: int: 1>**

Name eines Feldes, das mit FL definiert. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Anzahl Spalten interpretiert wird, hinterlegt.

**LEFTMAR=<integer 0..31>**

Für alle Drucker mit Ausnahme der RSO-Drucker: Der Ausgabetext soll um die angegebene Anzahl Spalten eingerückt werden.

Für alle RSO-Drucker:

Der Ausgabetext soll um die angegebene Anzahl \* 1/10 Zoll eingerückt werden.

**LEFTMAR=\*STD**

Die Voreinstellung ist in der SPOOL-Parameterdatei hinterlegt und kann mit dem Kommando oder der SPSEVE-Anweisung SHOW-SPOOL-PARAMETERS abgefragt werden; die Information ist dem Ausgabefeld PRINT-CMD-DEFAULTS:...LEFT-MARGIN=... zu entnehmen.

**LINEPP=\*UNCHANGED / <var: int: 2> / <integer 1..32767> / \*STD**

Legt fest, wieviel Zeilen (einschließlich Überschrift und Leerzeilen) auf eine Seite zu drucken sind.

**LINEPP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**LINEPP=<var: int: 2>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 2 Byte), die als Anzahl der Zeilen auf einer Seite interpretiert wird, hinterlegt.

**LINEPP=<integer 1..32767>**

Anzahl der Zeilen auf einer Seite.

**LINEPP=\*STD**

Wird kein Angabe gemacht, errechnet sich - unabhängig von der Angabe bei dem Operanden HEADLIN - die Zahl der Zeilen pro Druckseite nach folgender Formel:

$$\text{Zeilenanzahl} = P * Z - A - 6$$

Dabei bedeuten:

P = Papiergröße in Zoll

Z = Zeilendichte

A = Anzahl der Zeilen vor erstem Vertikaltabulator "Kanal 1"

*Drucker mit ladbarem Vorschubinformationspuffer*

- Der Vertikaltabulator "Kanal 1" legt die Zeile für den Druckbeginn fest. Standardmäßig werden 2 Leerzeilen vor Druckbeginn eingestellt, d.h. im Loop steht "Kanal 1" (CHANNEL 01) in der dritten Zeile.
- Ist der bei dem Operanden LINEPP angegebene Wert größer als die angegebene Zeilenanzahl im Loop, wird der im Loop vorgegebene Wert genommen.
- Ein hier angegebener Wert muss mindestens dreimal so groß sein wie der bei LINEPP=\*SPACE\_1 / \*SPACE\_2 / \*SPACE\_3 angegebene Zeilenvorschub, wenn der Operand LINEPP zusammen mit den Operanden HEADLIN und LINEPP angegeben wird.

**HEADLIN=\*UNCHANGED / <var: enum-of \_headline\_set:1> / \*NO / \*STD / \*DATE / \*FRECORD / \*PAGE / \*DATE\_FRECORD / \*DATE\_PAGE / \*FRECORD\_PAGE / \*DATE\_FRECORD\_PAGE**

Legt fest, ob auf jede Seite (ausgenommen Deck- und Schlussblatt) eine Überschriftzeile zu drucken ist.

**HEADLIN=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**HEADLIN=<var: enum-of \_headline\_set:1>**

Die Vereinbarung bzgl. der Überschriftszeile wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*NO
2	*STD
4	*FRECORD
8	*DATE
12	*DATE_FRECORD
16	*PAGE
20	*FRECORD_PAGE
24	*DATE_PAGE
28	*DATE_FRECORD_PAGE
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**HEADLIN=\*NO**

Eine Überschriftszeile wird nicht gedruckt.

**HEADLIN=\*STD**

Die Überschrift hat folgenden Aufbau:

Überschrift	DATE jjjj-mm-tt	userid	datei	PAGE nnnn
Spalte	1	41(11,11)	60(21,67)	124 (77)

Die erste Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge < 132 und ≥ 80 Zeichen (mit abweichenden Werten). Die zweite Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge < 80 und ≥ 70 Zeichen (mit abweichenden Werten).

jj-mm-tt	Jahr-Monat-Tag
userid	Benutzerkennung
datei	Dateiname
nnnn	Seitennummer

Wenn nicht anders vereinbart, folgt der Überschriftzeile eine Leerzeile.  
 Die Überschrift verschiebt sich entsprechend der Angabe bei dem Operanden LEFTMAR, die Überschriftzeile wird jedoch ab Spalte 132 abgeschnitten.  
 DATE und PAGE sind in der Überschriftenzeile nur vorhanden, wenn gilt:  
 Zeilengröße  $\geq$  032.

**HEADLIN=\*DATE / \*FRECORD / \*PAGE / \*DATE\_FRECORD / \*DATE\_PAGE / \*FRECORD\_PAGE / \*DATE\_FRECORD\_PAGE**

Die Überschrift hat folgenden Aufbau:

HEADLIN=	*DATE	*FRECORD	*PAGE
Überschrift	DATE jjjj-mm-tt (jjjj-mm-tt)	erster Satz	PAGE nnnn (nnnn, nnnn)
Spalte	1	21	124 (77,67)

Bei \*DATE:

Die Angabe in Klammern gilt für eine Formulardefinition mit einer Zeilenlänge  $<$  132 Zeichen und  $\geq$  70 (mit abweichenden Werten).

Bei \*PAGE

Die erste Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge  $<$  132 und  $\geq$  80 Zeichen (mit abweichenden Werten). Die zweite Angabe in Klammern gilt für eine Formulardefinition bei einer Zeilenlänge  $<$  80 und  $\geq$  70 Zeichen (mit abweichenden Werten).

jjjj-mm-tt      Jahr-Monat-Tag  
 erster Satz    erster logischer Satz der Datei  
 nnnn            Seitennummer

Fehlt eine der Angaben \*DATE, \*FRECORD oder \*PAGE , wird der entsprechende Abschnitt mit Leerzeichen aufgefüllt.

Wenn nicht anders vereinbart, folgt der Überschriftzeile eine Leerzeile.  
 Die Überschrift verschiebt sich entsprechend der Angabe bei dem Operanden LEFTMAR, die Überschriftzeile wird jedoch ab Spalte 132 abgeschnitten.  
 \*DATE und \*PAGE sind in der Überschriftenzeile nur vorhanden, wenn gilt:  
 Zeilengröße  $\geq$  032.

Bei Angabe von \*FRECORD wird der erste Satz nicht zum Datum zugehörig gesehen.  
 Die Angaben können durch "\_" (Unterstrich) verknüpft werden.

**OUTFORM=UNCHANGED / <var: enum-of \_outform\_set:1> / \*CHARACTER / \*HEXADECIMAL**

Gibt an, ob nur im Zeichenformat oder zusätzlich sedezimal auszugeben ist.

**OUTFORM=UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**OUTFORM=<var: enum-of \_outform\_set:1>**

Die Vereinbarung bzgl. der Ausgabe der Zeichen wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	*CHARACTER
2	*HEXADECIMAL
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**OUTFORM=\*CHARACTER**

Gibt nur im Zeichenformat aus. Datensätze, die die Druckzeilenlänge überschreiten, werden abgeschnitten.

**OUTFORM=\*HEXADECIMAL**

Gibt die Datensätze im Zeichenformat und in sedezimaler Darstellung aus.

*Ausgabeformat*

Am Anfang jeder Ausgabezeile steht ein 8 Byte langer Vorspann, auf ihn folgen die Daten in der Länge von 50 Byte.

Jede Ausgabezeile wird zuerst gemäß Font gedruckt und anschließend sedezimal wiederholt.

*Aufbau der Ausgabezeile*

Spalte	Inhalt
1-4	Spaltennummer, ab der die Daten des ausgegebenen Satzes beginnen
5-8	Leerzeichen
ab 9	Zeichen des Eingabesatzes gemäß Font; die einzelnen Zeichen sind durch ein Leerzeichen getrennt. In der nächsten Zeile werden die Zeichen in sedezimaler Form wiederholt.

**TWOSIDE=\*UNCHANGED / <var: enum-of \_twoside\_set:1> / \*STD / \*NO / \*YES / \*TUMBLE / \*IGNORE**

Legt fest, ob auf einem LP65-Drucker, PCL-Drucker, APA-Drucker bzw. dem 4830-Drucker einseitig oder doppelseitig gedruckt werden soll.

**TWOSIDE=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**TWOSIDE=<var: enum-of \_twoside\_set:1>**

Die Vereinbarung bzgl. des ein- oder doppelseitigen Druckens wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*STD
1	*YES
2	*TUMBLE
4	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

*Hinweis*

Die Funktionen TWOSIDE, OVERLAY und PAGECOP sind verknüpft und Teil ein und desselben Druckerkommandos. Wird nur einer dieser Operanden spezifiziert, werden für die übrigen Standardwerte generiert.

**TWOSIDE=\*STD**

Wie in der PCL-Datei definiert, wird einseitig (SIMPLEX-Modus) oder doppelseitig (DUPLEX-Modus) gedruckt.

**TWOSIDE=\*NO**

Es wird im SIMPLEX-Modus, d.h. einseitig, gedruckt.

**TWOSIDE=\*YES**

Es wird im DUPLEX-Modus, d.h. doppelseitig, gedruckt.

**TWOSIDE=\*TUMBLE**

Es wird im DUPLEX-Modus, d.h. doppelseitig, gedruckt, und die Seiten werden nicht von links nach rechts, sondern von oben nach unten umgedreht.

**TWOSIDE=\*IGNORE**

Die Einstellung für den Operanden TWOSIDE soll ignoriert werden.

**ROT=\*UNCHANGED / <var: enum-of \_rotation\_set: 1> / \*NO / \*ROT\_0 / \*ROT\_90 / \*ROT\_180 / \*ROT\_270 / \*ROT\_0\_180 / \*ROT\_90\_270 / \*ROT\_180\_0 / \*ROT\_270\_90 / \*BY\_CONTROL\_CODES**

Legt fest, ob die zu druckenden Seiten des SPOOL-OUT-Auftrags gedreht auszugeben sind; wenn ja, um wieviel Grad gedreht.

Für die Druckertypen 4830-PCL und 4850-PCL können nur die Werte 0, 90, 180 und 270 angegeben werden.

Die im Drucker aufgebaute Druckseite wird um eine bestimmte Gradzahl gedreht (im Uhrzeigersinn) und auf die Papiervorlage gedruckt; zum Beispiel kann hochkant eingelegtes Papier im Querformat bedruckt werden. Bei Seitendrehung um  $90^0/270^0$  wird ein gesonderter Loop benötigt (siehe Operand ROTLOOP). Die Ausgabe erfolgt bei Angabe des Operanden - außer bei ROT=\*NO - (automatisch) auf HP-Drucker und HP90-Drucker. SHOW-SPOOL-PARAMETERS informiert darüber, ob in der aktuellen SPOOL-Konfiguration ein HP-Drucker oder ein HP90-Drucker, PCL-Drucker sowie 4830-PCL- und 4850-PCL-Drucker mit Seitendrehmodul zur Verfügung steht oder nicht:

Ausgabefeld DEVICE-TYPE:..., ROT=YES/NO.

SPOOLOUT-Aufträge mit Seitendrehung werden in den Ausgaben für die Kommandos SHOW-USER-STATUS und SHOW-PRINT-JOB-STATUS JOB-IDENTIFICATION=TSN(TSN=...) angezeigt.

Der Vorschub für gedreht auszugebende Seiten wird prinzipiell über einen eigenen Loop gesteuert (ROTLOOP). Haben Sie weder einen ROTLOOP bei MODPJAT angegeben noch ein Formular mit definiertem ROTLOOP (siehe ADD-SPOOL-FORM), übernimmt der Standard-ROTATION-LOOP R06 im Standard-Formular die Vorschubsteuerung für die gedreht auszugebenden Seiten. Deck- und Schlussblätter werden prinzipiell nicht gedreht ausgegeben.

### **ROT=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

### **ROT=<var: enum-of \_rotation\_set: 1>**

Die Vereinbarung bzgl. der Seitendrehung wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

213	*NO
232	*BY_CONTROL_CODES
241	*ROT_90
242	*ROT_180
243	*ROT_270
244	*ROT_0_180
245	*ROT_90_270
246	*ROT_180_0
247	*ROT_270_90
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

### **ROT=\*NO**

Seitendrehung wird nicht ausgeführt. Eventuell in der Datei vorhandene Steuerzeichen für Seitendrehung werden nicht ausgewertet.



**ROT=\*ROT\_0 / \*ROT\_90 / \*ROT\_180 / \*ROT\_270**

Jede Druckseite wird (im Uhrzeigersinn) um  $90^0$  /  $180^0$  /  $270^0$  gedreht ausgegeben. In der Datei enthaltene Steuerzeichen für Seitendrehung werden nicht ausgewertet. Für Seitendrehung um  $90^0$  /  $270^0$  wird ein eigener Loop benötigt. Sie müssen dabei selbst prüfen, ob die Ausgabe mit dem angegebenen Loop nicht zu Fehlern führt. Im Fehlerfall wird der Auftrag abgewiesen.

**ROT=\*ROT\_0\_180 / \*ROT\_90\_270 / \*ROT\_180\_0 / \*ROT\_270\_90**

Die ungeraden Seiten (Zahl vor dem zweiten Bindestrich) und geraden Seiten (Angabe nach zweitem Bindestrich) sollen unterschiedlich gedreht ausgegeben werden.

**ROT=\*BY\_CONTROL\_CODES**

Steuerzeichen zur Seitendrehung in der Datei werden ausgewertet, wenn zugleich CONTMOD=\*PHYSICAL angegeben ist.

**INTRAY=\*UNCHANGED / <var: int: 1> / <integer 1..99> / \*STD / \*IGNORE / \*BY\_FORMAT (...)**

Legt das Papiereingabefach für LP65-Drucker, PCL-Drucker, APA-Drucker und die RSO-Drucker 2030-PCL, 4011, 4812, 4813, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-EPLQ, 9000-EPSQ, 9000-PCL, 9004, 9011, 9012, 9013, 9014, 9015, 9021, 9022-200, 9025, 9026-PCL, 9026-RENO, 9097 und DJET fest. Bei dem Druckertyp LP65 können die Fachnummern 1 bis 3, bei RSO-Druckern die Fachnummern 1 bis 99 angegeben werden.

**INTRAY=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**INTRAY=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Nummer des Papiereingabefaches interpretiert wird, hinterlegt.

**INTRAY=<integer 1..99>**

Nummer des Einzugsfaches, aus dem das Papier sowohl für die Ausgabe der Datei selbst als auch zum Drucken des Deckblattes (HEADER-PAGE) und des Schlussblattes (TRAILER-PAGE) genommen wird.

Der angegebene Wert wird von RSO nicht überprüft. Mögliche Werte: 1...9.

Die folgende Tabelle zeigt den Druckertyp und die jeweils maximale Anzahl der Einzugsfächer (Spalte „max.“).

Drucker	max.	Drucker	max.	Drucker	max.	Drucker	max.
2030-PCL	2	8121	0	9002	0	9022	2
4011	2	9000	0	9003	0	9022-200	2
4812	2	9000-PCL	2	9004	3	9025	2
4813	2	9000-PRO	0	9011	2	9026-PCL	4
4818-PCL	2	9000-PS	0	9012	2	9026-RENO	4
4821-PCL	2	9000-EPFX	0	9013	3	9045-ANSI	0
4822-PCL	3	9000-EPLQ	2	9014	3	9046	0
4824-PCL	2	9000-EPsq	2	9015	2	9645	0
4825-PCL	3	9001	0	9021	2	DJET	1
4830-PCL	3	9001-31	0				
4850-PCL	2	PCL-Drucker	3				

### **INTRAY=\*STD**

Falls ein Druckauftrag an den lokalen SPOOL oder an einen speziellen Druckerpool adressiert wird, wird dieser Operand ignoriert. Andernfalls, wenn der Druckauftrag z. B. an einen bestimmten RSO-Drucker adressiert wird, wird Papier von dem Papiereingabefach verwendet, das im Geräteeintrag des LP65-Druckers in der SPOOL-Parameterdatei definiert ist. Die Auswahl des Papiereingabefaches ist gültig für den gesamten SPOOL-OUT-Vorgang, d.h. einschließlich der Verarbeitung von Deck- und Schlussblatt.

### **INTRAY=\*IGNORE**

Nur für RSO.

Die Auswahl des Papiereingabefaches wird durch eine PCL-Datei oder über eine Eingabe an der Bedienungskonsole des Druckers gewählt.

Auch die PCL-Datei kann ihrerseits ein Papiereingabefach auswählen.

Die Prioritätsreihenfolge bei der Auswahl eines Papiereingabefaches ist wie folgt:

1. die im MODPJAT angegebene Nummer des Papiereingabefaches
2. die eventuell erfolgte Auswahl über die PCL-Datei, die im Makro MODPJAT angegeben wurde
3. die eventuell erfolgte Auswahl über die PCL-Datei, die in der Formulardefinition in der SPOOL-Parameterdatei festgelegt wurde.
4. die Standardwerte, die im Geräteeintrag definiert sind.

**INTRAY=\*BY\_FORMAT (...)**

Das Papiereingabefach wird über ein Schlüsselwort für das Format im Operanden INTRAYF festgelegt.

**INTRAYF=<var: enum-of \_inray\_format\_set:1> / \*MANUAL / \*A3 / \*A4 / \*A5 / \*B4 / \*B5 / \*DOUBLE\_LETTER / \*EXEC / \*FOLIO / \*INVOICE / \*LEGAL / \*LETTER / \*MONARCH / \*COMMERCIAL\_10 / \*DL / \*C5 / \*A3\_UNCUT / \*A4\_UNCUT / \*LEDGER**

Legt das Papiereingabefach über ein Schlüsselwort für das Format fest.

Um sicherzustellen, dass die im Operanden INTRAYF die angegebenen Werte für Papiergröße und Zeilenlänge die für das gewählte Papierformat zulässigen Maximalwerte nicht überschreiten, werden die Werte des INTRAYF-Operanden mit den zulässigen Maximalwerten für das Papierformat verglichen. Die folgende Tabelle enthält die für das jeweilige Papierformat zulässigen Maximalwerte.

Papierformat	Maximalwert für die Papiergröße	Maximalwert für die Zeilenlänge
A3	165	116
A4	116	82
A5	32	58
B4	143	101
B5	101	71
FOLIO	129	85
INVOICE	85	55
EXEC	105	722
LEGAL	140	85
LETTER	110	85
DOUBLE-LETTER	150	117
MONARCH	75	38
COMERCIAL-10	95	41
DL	86	43
C5	90	63
a3-uncut	120	120
a4-uncut	169	84
ledger	170	110

**INTRAYF=<var: enum-of \_inray\_format\_set:1>**

Die Vereinbarung bzgl. des Papiereingabefachs wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

128	MANUAL
129	A3
130	A4
131	A5
132	B4
133	B5
134	DOUBLE-LETTER
135	EXEC
136	FOLIO
137	INVOICE
138	LEGAL
139	LETTER
140	MONARCH
141	COMMERCIAL-10
142	DL
143	C5
144	A3_UNCUT
145	A4_UNCUT
146	LEDGER
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

Diese Werte sind in der ASSEMBLER-, C- und COBOL-Schnittstelle definiert. In der C-Schnittstelle steht darum z.B. Folgendes:

```

/* inray_format_set */
/* ENUM _inray_format_set */
<< Die Vereinbarung bzgl. des Eingabefachs wird >>
<< nicht explizit, sondern über folgende Konstantendefinition angegeben>>
<< (in der Parameterliste des Funktionsaufrufs kann darum der >>
<< in der Konstanten hinterlegte ganzzahlige Wert verwendet werden): >>
#define PRNTDOCinray_format_manual 128 /* inray_format = *manual */
#define PRNTDOCinray_format_a3 129 /* inray_format = *a3 */

```

```

#define PRNTDOCinray_format_a4 130          /* inray_format = *a4      */
#define PRNTDOCinray_format_a5 131          /* inray_format = *a5      */
#define PRNTDOCinray_format_b4 132          /* inray_format = *b4      */
#define PRNTDOCinray_format_b5 133          /* inray_format = *b5      */
#define PRNTDOCinray_format_double-letter 134
/* inray_format = *double-letter*/
#define PRNTDOCinray_format_exec 135        /* inray_format = *exec    */
#define PRNTDOCinray_format_folio 136       /* inray_format = *folio   */
#define PRNTDOCinray_format_invoice 137     /* inray_format = *invoice1 */
#define PRNTDOCinray_format_legal 138       /* inray_format = *legal   */
#define PRNTDOCinray_format_letter 139      /* inray_format = *letter  */
#define PRNTDOCinray_format_monarch 140     /* inray_format = *monarch */
#define PRNTDOCinray_format_commercial-10 141
/* inray_format = *commercial-10 */
#define PRNTDOCinray_format_d1 142          /* inray_format = *d1      */
#define PRNTDOCinray_format_c5 143          /* inray_format = *c5      */

```

**INTRAYF=\*MANUAL**

Die Papierzufuhr erfolgt von Hand, daher sollten Sie jedes Mal ein Blatt Papier einlegen, wenn eine neue Seite bedruckt werden soll bzw. der Drucker Sie dazu auffordert. Folgende Drucker können die manuelle Papierzufuhr unterstützen: 9014, 9021, 9022, 9022-200, 9026-Reno, 9026-PCL, 4812, DJET, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 9000-PCL.

**INTRAYF=\*A3**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das A3-Bögen enthält. Der Wert \*A3 ist für die Drucker 4830-PCL, 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*A4**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das A4-Bögen enthält. Der Wert \*A4 ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*A5**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das A5-Bögen enthält. Der Wert \*A5 ist für die Drucker 4830-PCL, 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*B4**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das B4-Bögen enthält. Der Wert \*B4 ist für die Drucker 4830-PCL und 9026-RENO anwendbar.

**INTRAYF=\*B5**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das B5-Bögen enthält. Der Wert \*B5 ist für die Drucker 4830-PCL, 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*FOLIO**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das FOLIO-Bögen enthält.

Der Wert \*FOLIO ist für den Drucker 9026-RENO anwendbar.

**INTRAYF=\*INVOICE**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das INVOICE-Bögen enthält.

Der Wert \*INVOICE ist für die Drucker 9026-PCL und 9026-RENO anwendbar.

**INTRAYF=\*EXEC**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das EXEC-Bögen enthält.

Der Wert \*EXEC ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*LEGAL**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das LEGAL-Bögen enthält.

Der Wert \*LEGAL ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*LETTER**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das LETTER-Bögen enthält.

Der Wert \*LETTER ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*DOUBLE-LETTER**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das DOUBLE-LETTER-Bögen enthält.

Der Wert \*DOUBLE-LETTER ist für den Drucker 9026-RENO anwendbar.

**INTRAYF=\*MONARCH**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das MONARCH-Bögen enthält.

Der Wert \*MONARCH ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL und 4825-PCL anwendbar.

**INTRAYF=\*COMMERCIAL-10**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das COMMERCIAL-10-Bögen enthält.

Der Wert \*COMMERCIAL-10 ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL und 4825-PCL anwendbar.

**INTRAYF=\*DL**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Umschläge der Größe International-DL enthält.

Der Wert \*DL ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*C5**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Umschläge der Größe International-C5 enthält.

Der Wert \*C5 ist für die Drucker 9021, 9000-PCL, 9026-PCL, 9026-RENO, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL und 4830-PCL anwendbar.

**INTRAYF=\*A3-UNCUT**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Bögen der Größe A3-UNCUT enthält.

Der Wert \*A3-UNCUT ist nur für PCL-Drucker anwendbar.

**INTRAYF=\*A4-UNCUT**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Bögen der Größe A4-UNCUT enthält.

Der Wert \*A4-UNCUT ist nur für PCL-Drucker anwendbar.

**INTRAYF=\*LEDGER**

Die Papierbögen werden aus dem ersten Eingabefach entnommen, das Bögen der Größe LEDGER enthält.

Der Wert \*LEDGER ist nur für PCL-Drucker anwendbar.

**OUTTRAY=\*UNCHANGED / <var: int: 1> / <integer 1..99> / \*STD / \*IGNORE / \*SORTER**

Legt das Papierausgabefach für LP65-Drucker, APA-Drucker und für die RSO-Drucker 2030-PCL, 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 4830-PCL, 4850-PCL, 9000-PCL, 9014, 9015, 9026-PCL und 9026-RENO fest.

Der Auftrag kann nur auf Druckern mit Einzelblattverarbeitung ausgeführt werden.

**OUTTRAY=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**OUTTRAY=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Nummer des Papierausgabefaches interpretiert wird, hinterlegt.

**OUTTRAY=<integer 1..99>**

Mit diesem Wert legen Sie fest, welches Papierausgabefach für den aktuellen Auftrag benutzt wird.

**OUTTRAY=\*STD**

Die Ausgabe erfolgt in dem Papierausgabefach, das im Geräteeintrag (SPOOL-Parameterdatei) als Standard-Papierausgabefach definiert wurde.

Die Auswahl des Papierausgabefaches ist gültig für den gesamten SPOOL-OUT-Vorgang, d.h. einschließlich der Verarbeitung von Deck- und Schlussblatt.

Die Auswahl des Papierausgabefaches durch das BS2000 kann verhindert werden durch eine PCL-Datei oder über eine Eingabe an der Bedienungskonsole des Druckers. In diesen Fällen haben die Angaben bei OUTTRAY keine Auswirkungen. Auch in der PCL-Datei kann ein Papierausgabefach definiert werden.

Die Prioritätsreihenfolge bei der Auswahl eines Papierausgabefaches ist wie folgt:

1. die im MODPJAT angegebene Nummer des Papierausgabefaches, soweit dies nicht verhindert wird durch eine PCL-Datei oder eine Eingabe an der Druckerkonsole.
2. die eventuell erfolgte Auswahl über die PCL-Datei, die in MODPJAT angegeben wurde.
3. die eventuell erfolgte Auswahl über die PCL-Datei, die in der Formulardefinition in der SPOOL-Parameterdatei festgelegt wurde.
4. die Standardwerte, die im Geräteeintrag definiert sind.

**OUTTRAY=\*IGNORE**

Wenn Sie diesen Wert angeben, sendet die Druckersteuerung keinen Code zur Auswahl eines Ausgabefaches an den Drucker. Dadurch können Sie das Ausgabefach in der Prologdatei festlegen.

**OUTTRAY=\*SORTER**

Mit dieser Angabe legen Sie fest, dass der Sortiermechanismus für den aktuellen Auftrag benutzt wird; und zwar mit Hilfe des Operanden SRTMODE. Der Sortiermechanismus ist zulässig für die Drucker 9026-RENO, 4818-PCL, 4822-PCL und 4825-PCL. Dieser Sortiermechanismus kann für bis zu 20 Papierausgabefächer eingesetzt werden. Er kann nicht durch die Standardwerte kontrolliert werden, die im Geräteeintrag definiert sind.

*Hinweis*

Druckerressourcen wie PROLOG-, EPILOG-, DIA-, MEMBER und Fontzeichen-Dateien werden vor der Auswahl der Papierfachausgabe gesendet.

**COVPAGE=(headtxt, headexi, traiexi)**

**headtxt:** <var: char: 32> / <c-string 1..32: c-string\_1..32> / \*NONE

Legt fest, ob eine Zeichenfolge auf Deck- und Schlussblättern erscheinen soll.

**headtxt:** <var: char: 32>

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 32 Byte), die als zu speichernde Information interpretiert wird, hinterlegt.



**headtxt: <c-string 1..32: c-string\_1..32>**

Die angegebene Information (max. 32 Zeichen) wird zur Verarbeitung von Exits im SCB gespeichert. Die ersten 8 Zeichen werden auf das Deckblatt als Großdruckzeile unter die Mailing Box gedruckt.

Nur Buchstaben, Ziffern und einige Sonderzeichen werden - als Großdruck - dargestellt, alle anderen Zeichen werden automatisch durch das abdruckbare Zeichen '?' für das Deckblatt ersetzt.

*Aufbau des Deckblattes*

1. Benutzerkennung in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
2. Abrechnungsnummer in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
3. Auftragsname in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
4. Mailing Box (Adress- und Identifikationsfeld; 12 Zeilen + 2 Leerzeilen)
5. 'text' in "Riesen-Buchstaben" (10 Zeilen + 2 Leerzeilen)

*Prioritätsreihenfolge*

1. Mailing Box (Adress- und Identifikationsfeld; 12 Zeilen + 2 Leerzeilen)
2. 'text' in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
3. Auftragsname in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
4. Benutzerkennung in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)
5. Abrechnungsnummer in „Riesen-Buchstaben“ (10 Zeilen + 2 Leerzeilen)

**headtxt: \*NONE**

Es soll keine Zeichenfolge auf Deck- und Schlussblättern erscheinen.

**headexi: \*UNCHANGED / <var: int: 4> / <integer 0..2147483639> / \*NO**

Nummer der gewünschten Deckblätter.

Die genaue Bedeutung des Operanden hängt jeweils von den Rechenzentrumspezifischen Vereinbarungen ab.

**headexi: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**headexi: <var: int: 4>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 4 Byte), die als Nummer der Deckblätter interpretiert wird, hinterlegt.

**traixi: \*UNCHANGED / <var: int: 4> / <integer 0..2147483639> / \*NO**

Nummer der gewünschten Schlussblätter.

Die genaue Bedeutung des Operanden hängt jeweils von den Rechenzentrumspezifischen Vereinbarungen ab.

**traixi: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**traixi: <var: int: 4>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 4 Byte), die als Nummer der Schlussblätter interpretiert wird, hinterlegt.

**FORM=\*UNCHANGED / <var: char: 6> / <c-string 1..6: alphanum\_name\_1..6> / \*STD**

Bezeichnet das Papierformat, das für die Ausgabe verwendet werden soll (z.B. STD, STDSF1, STDWA4).

**FORM=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**FORM=<var: char: 6>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 6 Byte), die als Name des Formulars interpretiert wird, hinterlegt.

**FORM=<c-string 1..6: alphanum\_name\_1..6>**

Name des Formulars, mit dem der SPOOLOUT-Auftrag verarbeitet werden soll. Implizit wird mit der Formularangabe ein Loop (oder eine Seiten- und Formatdefinition für APA-Drucker) benannt. Der zugeordnete Loop (bzw. die PAGEDEF und FORMDEF) muss in einer bestimmten Druckersteuerdatei enthalten sein. Die folgende Tabelle gibt Aufschluss darüber, welche Druckersteuerdatei für den jeweiligen Druckertyp diesen Loop (bzw. die PAGEDEF und FORMDEF) enthalten muss:

Ausgabegerät	Druckersteuerdatei mit dem Loop für das angegebene Formular
Drucker 3337, 3338, 3339, 3348, 3349, 3365, LP-EMULATED	\$\$SYSSPOOL.PRFILE
Drucker: 3351, 3353, 2090, 2140, 2240	\$\$SYSSPOOL.PRFILE oder die im Operanden USERRES angegebene Benutzer-PRFILE
APA-Drucker	\$\$SYSSPOOL.SYSPRT.SPS.021 oder die im Operanden USERRES angegebene Benutzer-SPSLIB

Der über den FORM-Operanden implizit benannte Loop wird ignoriert, wenn zugleich der Operand LOOP angegeben wird.

Ohne die Operanden FORM und LOOP wird mit dem für den jeweiligen Druckertyp eingetragenen Standardformular gedruckt.

Ein bei dem Operanden LOOP explizit angegebener Loop muss die gleiche Länge haben wie der dem verwendeten Formular zugeordnete Loop.

Bei APA-Druckern können keine Loops angegeben werden. Wenn Seiten- und Formatangaben im FORM-Operanden gemacht werden, werden diese zum Drucken von Header, Trailer und Meldungsseiten benutzt.

**FORM=\*STD**

Standardformular.

**LOOP=\*UNCHANGED / <var: char: 3> / <c-string 1..3: alphanum\_name\_1..3> / \*STD**  
 Name des Loops, der in den Vorschubinformationspuffer (VFB) geladen werden soll.  
 Der Loop-Name darf die Zeichen '\$', '&' und '@' nicht enthalten.

**LOOP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**LOOP=<var: char: 3>**

Name des Loops, der den Vorschub steuern soll. Die Länge des angegebenen Loops muss mit der Länge des Standard-Loops des verwendeten Formulars übereinstimmen.  
 Ein Loop zur Vorschubsteuerung wird benötigt für die HP- und HP90-Drucker (3351, 3353, 2090, 2140) und Drucker der Typen 3337, 3338, 3339, 3348, 3349, 3365.  
 Loops sind gespeichert in der Druckersteuerdatei PRFILE. Wird kein Loop angegeben, werden die impliziten Angaben bei dem Operanden FORM benutzt. Ohne Angabe der Operanden FORM oder LOOP werden Standardwerte eingesetzt.

**LOOP=<c-string 1..3: alphanum\_name\_1..3>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 3 Byte), die als Name des Loops interpretiert wird, hinterlegt.

**LOOP=/ \*STD**

Die Vorschubsteuerung für den SPOOL-OUT-Auftrag soll mit dem Standard-Loop des verwendeten Formulars realisiert werden.

**ROTLOOP=\*UNCHANGED / <var: char: 3> / <c-string 1..3: alphanum\_name\_1..3> / \*STD**

Nur für HP-Drucker und HP90-Drucker mit Seitendrehmodul.  
 Loop, mit dem die Ausgabe im Querformat gesteuert werden soll.  
 Der Loop-Name darf die Zeichen '\$', '&' und '@' nicht enthalten.

**ROTLOOP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**ROTLOOP=<var: char: 3>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 3 Byte), die als Name des Loops interpretiert wird, hinterlegt.

**ROTLOOP=<c-string 1..3: alphanum\_name\_1..3>**

In Hochkommata einzuschließende Zeichenfolge für den Namen des Loops, mit dem der Vorschub für die gedreht auszugebenden Seiten gesteuert werden soll.

**ROTLOOP=\*STD**

Die Ausgabe gedreht auszugebender Seiten soll über den Standard-ROTATION-Loop des angegebenen Formulars bzw. über den Standard-ROTATION-Loop R06 des Standardformulars (wenn im Makro MODPJAT kein Formular angegeben wurde) gesteuert werden.

**CHARSET=\*UNCHANGED / array(16): <var: char: 3> /****<c-string 1..3: alphanum\_name\_1..3> / \*STD**

Namen der Fonts, die für die Ausgabe zu verwenden sind.

In der Liste können für lokalen SPOOL maximal 4, für RSO maximal 16 Fonts angegeben werden. Die Fonts müssen für lokalen SPOOL in der Ressourcen-Bibliothek \$SYSSPOOL.PRFILE oder in einer Anwender-PRFILE (die mit dem Operanden USERRES anzugeben ist) enthalten sein.

Werden mehr als 4 Fonts benötigt, dann muss ein Font-Pool (Operand POOLNAM) angegeben werden.

Die Verwendung mehrerer Fonts in einem SPOOL-OUT-Auftrag setzt die Angabe CONTMOD=\*PAGE\_MODE voraus. Bei DOCFORM=\*TEXT wird nur der erste angegebene Font zum Drucken der (gesamten) Datei verwendet. Der Name des ersten Fonts und die Anzahl der angegebenen Fonts wird in der Ausgabe für das Kommando SHOW-PRINT-JOB-STATUS JOB-IDENTIFICATION=TSN(TSN=...) angezeigt.

Nur FUJITSU-SIEMENS-Standardfonts dürfen einen Namen bestehend aus drei numerischen Zeichen tragen.

Ist der Operand HEADLIN angegeben, so wird für die Überschriftzeile der erste angegebene Font verwendet.

Die Voreinstellung für CHARSET für das verwendete Formular kann mit SHOW-SPOOL-FORMS abgefragt werden; die Information ist dem Ausgabefeld C-S zu entnehmen.

Bei HP-Druckern ist zu beachten:

Die Systembetreuung kann einstellen, ob das Deckblatt mit dem Standardfont oder mit dem bei CHARSET angegebenen Font gedruckt wird. Diese Voreinstellung kann mit dem Kommando bzw. der SPSEVE-Anweisung SHOW-SPOOL-PARAMETERS abgefragt werden; die Information ist dem Ausgabefeld HEADER-PAGE: CHARACTER-SET=... zu entnehmen.

**CHARSET=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**CHARSET=<var: char: 3>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 3 Byte), die als Name des Fonts interpretiert wird, hinterlegt.

**CHARSET=<c-string 1..3: alphanum\_name\_1..3>**

Namen der Fonts, mit denen der SPOOL-OUT-Auftrag abgearbeitet werden soll. Die Zeichenfolge ist in Hochkommata einzuschließen.

Bei DOCFORM=\*TEXT (Voreinstellung) wird nur der erste angegebene Font zum Drucken verwendet.

Für Laserdrucker im lokalen SPOOL-Betrieb können maximal 4 Fonts angegeben werden. Die Druckersteuerzeichen zur Fontumschaltung werden nur ausgewertet, wenn zugleich CONTMODE=\*PAGE\_MODE angegeben wird.

Für RSO-Drucker können maximal 16 Fonts angegeben werden. Fontkennzeichen (CSI) im Text werden nur ausgewertet, wenn zugleich CONTMOD=\*LOGICAL angegeben wird. Der Makro wird abgewiesen, wenn in Verbindung mit DOCFORM=\*TEXT eine Liste mit Fonts angegeben wird.

### **CHARSET=\*STD**

Aus dem gewünschten Formular wird der Standardfont für diesen Druckertyp gewählt. Er kann mit SHOW-SPOOL-FORMS abgefragt werden.

### **POOLNAM=\*UNCHANGED / <var: char: 4> / <c-string 1..4: name\_1..4> / \*NONE**

Namen des Font-Pools (nur für HP-Drucker und HP90-Drucker), der für die Ausgabe zu verwenden ist.

### **POOLNAM=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

### **POOLNAM=<var: char: 4>**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 4 Byte), die als Name des Font-Pools interpretiert wird, hinterlegt.

### **POOLNAM=<c-string 1..4: name\_1..4>**

Bezeichnet einen Font-Pool (mit maximal 64 Fonts bei einem HP-PRINTER, maximal 46 Fonts bei einem HP90-PRINTER), aus dem ein oder mehrere Fonts zur Ausgabe verwendet werden sollen. Alle Fonts dieses Pools werden bei Ausführung des SPOOLOUT-Auftrags in den Fontpuffer geladen.

### **POOLNAM=\*NONE**

Es wird kein Font-Pool für die Ausgabe benötigt.

### **POOLIND=\*UNCHANGED / <var: int: 1> / <integer 0..64>**

Nummer des Fonts aus dem Font-Pool, mit dem der SPOOLOUT-Auftrag abgearbeitet werden soll. Die Nummer des Fonts wird durch seine Stellung bei der Definition des Font-Pools bestimmt. Der angegebene Font wird verwendet, wenn POOLIND zusammen mit CONTMOD=\*PAGE\_MODE angegeben wird.

### **POOLIND=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

### **POOLIND=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Nummer des Fonts interpretiert wird, hinterlegt.

### **POOLIND=<integer 0..64>**

Ganzzahliger Wert für die Nummer des Fonts aus dem Font-Pool.

### **EFO=\*UNCHANGED / <var: char: 2> / <c-string 2..2: c-string\_2..2> / \*NONE**

Legt fest, ob ein Filmdia - für HP90- und 3365 Drucker ein EFO-Datendia - zu verwenden ist, wenn ein SPOOLOUT-Auftrag verarbeitet werden soll.

**EFO=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**EFO=<var: char: 2>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 2 Byte), die als Name des Filmdias interpretiert wird, hinterlegt.

**EFO=<c-string 2..2: c-string\_2..2>**

Name des Filmdias (HP90 und 3365: EFO-Datendias), das zur Verarbeitung des SPOOLOUT-Auftrags zu verwenden ist (der Name muss mit der Systembetreuung vereinbart worden sein).

**EFO=\*NONE**

Es wird kein Filmdia (HP90 und 3365: kein EFO-Datendia) zur Ausgabe verwendet.

**OVERLAY=(*face*, *reverse*)**

Legt fest, ob ein EFO-Datendia für Drucker des Typs LP65 auf Vorder- und/oder Rückseite verwendet werden soll.

Sie müssen im Druckerspeicher vorhanden sein.

Die Operanden OVERLAY, TWOSIDE und PAGECOP sind Teil ein und desselben LP65-Druckerbefehls und insofern miteinander verknüpft. Wird nur der Operand OVERLAY spezifiziert, generiert SPOOL Standardwerte für die beiden anderen Operanden. Diese Standardwerte werden von jedem in einer PCL-Datei oder an der Druckerbedienungskonsole angegebenen Wert überschrieben.

Eine Übersicht über die möglichen Kombinationen dieser drei verknüpften Funktionen finden Sie in der Beschreibung des Druckertyps LP65 im Handbuch „[SPOOL \(BS2000/OSD\)](#)“.

***face*: <var: int: 1> / <integer 1..127> / \*STD / \*NONE**

Identifikationsnummer des Dias, das auf der Vorderseite verwendet werden soll.

***face*: <var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Identifikationsnummer interpretiert wird, hinterlegt.

***face*: <integer 1..127>**

Ganzzahliger Wert für die Identifikationsnummer des Dias, das auf der Vorderseite verwendet werden soll.

***face*: \*STD**

Der voreingestellte Wert für die Identifikationsnummer des Dias (bzw. \*NONE) ist in der SPOOL-Parameterdatei hinterlegt.

***face*: \*NONE**

Es soll kein Dia auf der Vorderseite verwendet werden.

***reverse* : <var: int: 1> / <integer 1..127> / \*STD / \*NONE**

Identifikationsnummer des Dias, das auf der Rückseite verwendet werden soll.

**reverse : <var: int: 1>**

Name eines Feldes, das mit FL definiert. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Identifikationsnummer interpretiert wird, hinterlegt.

**reverse : <integer 1..127>**

Ganzzahliger Wert für die Identifikationsnummer des Dias, das auf der Rückseite verwendet werden soll.

**reverse : \*STD**

Der voreingestellte Wert für die Identifikationsnummer des Dias (bzw. \*NONE) ist in der SPOOL-Parameterdatei hinterlegt.

**reverse : \*NONE**

Es soll kein Dia auf der Rückseite verwendet werden.

**FOB=\*UNCHANGED / <var: char: 4> /**

**<c-string 1..4: alphanumname\_1..4, c-string\_1..4> / \*NONE**

Legt fest, ob ein FOB-Datendia für die Verarbeitung des SPOOLOUT-Auftrags zu verwenden ist.

**FOB=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**FOB=<var: char: 4>**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 4 Byte), die als Name des Dias interpretiert wird, hinterlegt.

**FOB=<c-string 1..4: alphanumname\_1..4, c-string\_1..4>**

Name des Dias, der zur Verarbeitung des SPOOLOUT-Auftrags verwendet werden soll. Bei Angabe eines Dias wird die Datei auf HP-PRINTER oder HP90-PRINTER ausgegeben. Die Verwendung eines Dias für den SPOOLOUT-Auftrag wird in der Ausgabe des Kommandos SHOW-PRINT-JOB-STATUS JOB-IDENTIFICATION =TSN(TSN=...) angezeigt.

**FOB=\*NONE**

Es wird kein FOB-Datendia zur Ausgabe verwendet.

**PAGEPCL=\*UNCHANGED / <var: int: 4> / <integer 1..50000> / \*STD**

Gibt an, welche Seiten-Definition für den Ausdruck auf LP65- oder APA-Druckern benutzt werden soll.

**PAGEPCL=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PAGEPCL=<var: int: 4>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 4 Byte), die als Nummer der PCL-Datei interpretiert wird, hinterlegt.

**PAGEPCL=<integer 1..50000>**

Nur für LP65-Drucker:

Ganzzahliger Wert für die Nummer der PCL-Datei, mit der die Druckdatei ausgegeben werden soll.

**PAGEPCL=\*STD**

Für APA-Drucker:

Die in der SPSLIB festgelegte Standarddefinition soll benutzt werden; siehe dazu das Handbuch „[PRISMAproduction/BS2000 - SPS](#)“.

Für LP65-Drucker:

Nummer der PCL-Datei, mit der die Druckdatei ausgegeben werden soll. Nur die Druckdatei selbst wird mit dieser PCL-Datei ausgedruckt. Deck- und Schlussblatt werden über die PCL-Datei gesteuert, die im Formular der SPOOL-Parameterdatei definiert ist.

*Hinweis*

Eine PCL-Datei, die es unmöglich macht, eine andere PCL-Datei über ein Kanalkommando auszuwählen, darf nicht verwendet werden.

Nach dem Starten der angegebenen PCL-Datei wird der Betriebsmodus des Druckers überprüft. Ist der EXCCW-Modus aktiv und tritt während des Druckvorganges ein Fehler auf, ist der Wiederaufsetzpunkt der letzte SECTION-Satz; wurde im Makro MODPJAT keine Angabe bei dem Operanden SECTREC gemacht, wird die Datei von Anfang an erneut ausgedruckt.

**USERRES=\*UNCHANGED / <var: char: 44> /****<c-string 1..44: filename\_without-userid\_1..44, c-string\_1..44> / \*STD**

Bezeichnet eine Benutzerdatei, die alle benötigten Ressourcen für den Ausdruck auf unterschiedlichen Druckertypen enthält: selbsterstellte Loops, Fonts, Dias, Font-Pools, Code-Umsetzungstabellen und SPS Data Stream Definitionen. Es kann angegeben werden:

- eine Benutzer-PRFILE, die Loops, Fonts, Dia-Einträge (FOB) und Font-Pool-Einträge (Operand CHARSET) enthält.
- eine Benutzer-SPSLIB, die PAGEDEFS (Operand PAGEPCL), FORMDEFS, Fonts, Page-Segments, Dias und Raster-Bild-Daten enthält.
- Eine Benutzer-RSOFIL (nur für RSO-Betrieb), die Loops enthält.

Fehlt die Angabe einer Benutzer-PRFILE, -SPSLIB bzw. -RSOFIL, werden die entsprechenden Angaben der Datei \$SYSSPOOL.PRFILE, \$SYSSPOOL.SYSPRT.SPS.021 bzw. \$SYSSPOOL.RSOFIL entnommen.

**USERRES=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.



**USERRES=<var: char: 44>**

Name eines Feldes, das mit CL definiert ist oder ein Register, in dem der Wert steht. In diesem Feld bzw. dem Register wird eine Zeichenfolge (Länge 44 Byte), die als Name der Ressourcen-Datei interpretiert wird, hinterlegt.

**USERRES=<c-string 1..44: filename\_without-userid\_1..44, c-string\_1..44>**

Name einer Benutzer-PROFILE, -SPSLIB oder -RSOFILE, die eventuell eine Katalogkennung und eine Benutzerkennung enthält. Von SPOOL wird dieser Dateiname mit dem Suffix '.PROFILE', '.SPSLIB' oder '.RSOFILE' gebraucht.

Der String darf maximal aus 28 Zeichen (ohne Katalogkennung und Benutzerkennung) bestehen, um sicherzustellen, dass diese Benutzerdatei von jeder Kennung aus aufgerufen werden kann.

Wird der Dateiname ohne Benutzerkennung angegeben, wird die Datei erst unter der Benutzerkennung des Aufrufers, anschließend unter SYSSPOOL gesucht. Wird die Datei nicht gefunden, wird das Kommando abgewiesen.

Wird eine Benutzerkennung mitangegeben, wird nur unter dieser nach der Datei gesucht.

*Beispiel 1*

```
MODPJAT DNAME=(DATEI),USERRES='$XX.XX'
```

Es wird nach der Datei \$XX.XX.PROFILE gesucht. Wird die Datei nicht gefunden, so wird das Kommando abgewiesen.

*Beispiel 2*

Ist die Katalogkennung angegeben, wird die Suche auf den angegebenen Pubset beschränkt:

```
MODPJAT DNAME=(DATEI), USERRES=':A:XXXXX'
```

Die Datei :A:\$userid.XXXXX.PROFILE wird gesucht. Wird die entsprechende Datei nicht gefunden, wird weitergesucht nach der Datei :A:\$SYSSPOOL.XXXXX.PROFILE. Wird diese Datei nicht gefunden, wird das Kommando abgewiesen.

*Beispiel 3*

```
MODPJAT DNAME=(DATEI), USERRES=':A:$XX.XXXXX'
```

Die Datei :A:\$XX.XXXXX.PROFILE wird gesucht. Wird die Datei nicht gefunden, wird das Kommando abgewiesen.

Befindet sich die Datei auf einem PVS, der exportiert wurde (Kommando EXPORT-PUBSET), werden alle SPOOLOUT-Aufträge, die ihn benötigen, in die KEEP-Warteschlange gesetzt. Ist der PVS wieder verfügbar (Kommando IMPORT-PUBSET), werden die SPOOLOUT-Aufträge erneut gestartet.

**USERRES=\*STD**

Benötigte Ressourcen werden der Datei \$SYSSPOOL.PROFILE, \$SYSSPOOL.SYSPRT.SPS.021 oder \$TSOS.RSOFILE entnommen.

**TRANTAB=(name, file)**

Legt fest, ob zur Verarbeitung des SPOOLOUT-Auftrags eine Code-Umsetztabelle zu verwenden ist und aus welcher Datei diese ggf. zu entnehmen ist. Die Code-Umsetztabelle ist dann notwendig, wenn das Standard-Umschaltzeichen 'FF' durch ein beliebiges Zeichen ersetzt werden soll.

**name: <var: char: 8> / <c-string 1..8: alphanum\_name\_1..8, c-string\_1..8> / \*NONE**

Legt fest, ob zur Verarbeitung des SPOOLOUT-Auftrags eine Code-Umsetztabelle zu verwenden ist.

**name: <var: char: 8>**

Name eines Feldes, das mit CL definiert. In diesem Feld wird eine Zeichenfolge (Länge 8 Byte), die als Name der Code-Umsetztabelle interpretiert wird, hinterlegt.

**name: <c-string 1..8: alphanum\_name\_1..8, c-string\_1..8>**

Der Name der Code-Umsetztabelle, die zur Verarbeitung des SPOOLOUT-Auftrags verwendet werden soll, wird über eine Zeichenfolge in Hochkommata angegeben.

**name: \*NONE**

Es wird keine Code-Umsetztabelle verwendet.

**file: <var: char: 44> / <c-string 1..44: filename\_without-userid\_1..44, c-string\_1..44> / \*SYSTEM**

Legt fest, aus welcher Datei die Code-Umsetztabelle zu entnehmen ist.

**file: <var: char: 44>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 44 Byte), die als Name der Datei interpretiert wird, hinterlegt.

**file: <c-string 1..44: filename\_without-userid\_1..44, c-string\_1..44>**

Der Name der Datei, aus der die angegebene Code-Umsetztabelle stammt, wird über eine Zeichenfolge in Hochkommata angegeben.

**file: \*SYSTEM**

Die angegebene Code-Umsetztabelle wird standardmäßig der Datei \$SYSSPOOL.PRFILE entnommen; für RSO-Drucker kann auch eine Benutzer-RSOFILe angegeben werden.

**PRNAME=\*UNCHANGED / <var: char 1..8> / <c-string: c-string 1..8> / \*STD / \*IPP**

Spezifiziert den angeforderten Zieldrucker des Druckauftrags. Angegeben werden kann ein verteilter lokaler Druckerpool, ein nicht-verteilter lokaler Druckerpool, ein RSO-Druckerpool oder ein RSO-Drucker. Mit Distributed Print Services können Sie weitere Zieldrucker angeben. Näheres entnehmen Sie bitte dem Handbuch „[Distributed Print Services \(BS2000/OSD\)](#)“.

*Bei Angabe eines Druckerpools*

Die Ausgabe erfolgt auf einen beliebigen Drucker in dem angegebenen Geräte-Pool. Der Pool muss in der SPOOL-Parameterdatei definiert sein und kann maximal 16 RSO-Geräte oder 16 lokale SPOOL-Geräte enthalten, nicht jedoch beides gleichzeitig.

Geräte-Pools werden verwaltet mit den SPERVE-Anweisungen ADD-, MODIFY-, REMOVE-, SHOW-PRINTER-POOL, siehe Handbuch „[SPERVE \(BS2000/OSD\)](#)“.

Der SPOOL-OUT-Auftrag wird abgewiesen, wenn

- dem im Makro MODPJAT angeforderten Formular nicht mindestens ein Druckertyp aus dem Geräte-Pool zugewiesen ist (siehe Kommando oder SPERVE-Anweisung SHOW-SPOOL-FORMS).
- der Operand OVERLAY angegeben ist.

Bitte berücksichtigen Sie, dass der Pool Geräte verschiedener Typen enthalten kann: Enthält die auszudruckende Datei Steuerzeichen, die nur von einem bestimmten Druckertyp interpretiert werden, sollte ein Drucker dieses Typs im Makro MODPJAT (implizit) angegeben werden. Eine Möglichkeit besteht darin, im Makro MODPJAT ein Formular anzugeben, das nur für den gewünschten Druckertyp definiert ist (ADD-SPOOL-FORM).

**PRNAME=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PRNAME=<var: char 1..8>**

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 8 Byte), die als Name des Druckers interpretiert wird, hinterlegt.

**PRNAME=<c-string: c-string 1..8>**

Nur für RSO.

Symbolischer Name des RSO-Geräts, auf dem der SPOOL-OUT-Auftrag verarbeitet werden soll.

**PRNAME=\*STD**

Der SPOOL-OUT-Auftrag soll auf dem in den SPOOL-Parametern festgelegten (PRINT-CMD-DEFAULTS) Standardgerätetyp verarbeitet werden.

**PRNAME = \*IPP**

Nur für RSO

Der SPOOL-OUT-Auftrag soll auf einem IPP-Drucker ausgeführt werden, der durch seine URL adressiert wird.

**PRTYPE=\*UNCHANGED / <var: enum-of \_prtype\_set:1> / \*ANY / \*LP65\_PRINTER / \*HP\_PRINTER / \*APA\_PRINTER**

Gibt an, welcher Druckertyp den Druckauftrag verarbeiten soll. Nur lokale Druckertypen dürfen angegeben werden.

**PRTYPE=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PRTYPE=<var: enum-of \_prtype\_set:1>**

Die Vereinbarung bzgl. des Druckertyps wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	*ANY
1	*LP65_PRINTER
2	*APA_PRINTER
3	*HP_PRINTER
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

**PRTYPE=\*ANY**

Vom Benutzer wird kein spezieller Druckertyp angefordert. In diesem Fall bestimmt das Subsystem SPOOL automatisch die erlaubten Druckertypen, die die Benutzeranforderung bearbeiten können. Für die Ausgabe auf RSO-Drucker und Drucker in UNIX-Systemen muss \*ANY angegeben werden.

**PRTYPE=\*LP65\_PRINTER**

Der SPOOLOUT-Auftrag soll auf einem Drucker vom Typ LP65 verarbeitet werden.

**PRTYPE=\*HP\_PRINTER**

Der SPOOLOUT-Auftrag soll auf einem Drucker vom Typ HP bzw. HP90 verarbeitet werden. Die Angabe \*HP\_PRINTER schließt die Druckertypen 2090/2140/2240(HP90) mit ein. Die Steuerzeichen für Laserdrucker vom Typ HP und HP90 sind identisch; ein SPOOLOUT-Auftrag für einen HP-PRINTER kann auf einem HP90-PRINTER verarbeitet werden und umgekehrt.

Bei der Auswahl des Druckertyps kommt es darauf an, ob zugleich der Operand CONTMOD angegeben wird oder nicht:

Druckerwahl	CONTMOD=*PAGE_MODE (Voreinstellung)	CONTMOD=(*PAGE_MODE,,*HP)
<b>PRTYPE=* ANY</b>	Die Ausgabe kann auf allen Druckertypen erfolgen.	Die Ausgabe kann auf HP- bzw. HP90-Drucker erfolgen. Andere Steuerzeichen werden in die HP-/HP90-Steuerzeichen umgewandelt.
	Für HP-/HP90-Drucker gilt außerdem: Umwandlung der OVERPRINT-Funktion in die LINE-MERGE-Funktion.	
<b>PRTYPE=*HP_PRINTER</b>	Die Ausgabe kann nur auf HP- und HP90-Druckern verarbeitet werden. Die OVERPRINT-Funktion wird in die LINE-MERGE-Funktion umgewandelt.	
	Einschränkungen PRFILE: Die Datei darf nicht das Zeichen X'FF'enthalten.	Unterstützung der HP-spezifischen Steuerzeichen. Einschränkung: Die Datei darf das Zeichen X'FF' nur als Umschaltzeichen enthalten, wenn die Steuerzeichenauswertung eingeschaltet ist.

#### *Hinweis für HP-/HP90-Drucker*

Wenn die Angaben im MODPJAT eine Ausgabe auf HP- bzw. HP90-Laserdrucker bestimmen, muss eine PRFILE zur Verfügung stehen. Andernfalls wird das Kommando abgewiesen. Steht in einer Installation kein HP/HP90 zur Verfügung oder nur Geräte mit ungenügendem Ausbau (z. B. zu wenige Fonts, kein Grafikspeicher für Dias, kein Seitendrehmodul), können die Aufträge nur auf Replay-Band ausgegeben werden. Die Systembetreuung kann sich mit dem Kommando SHOW-SPOOL-JOB-STATUS über solche Aufträge informieren.

#### **PRTYPE=\*APA\_PRINTER**

Der SPOOLOUT-Auftrag soll auf einem 2050-APA-PRINTER, einem 2090-APA-PRINTER oder einem 2090-TWIN-PRINTER verarbeitet werden.

#### **SRTMODE=\*UNCHANGED / <var: enum-of \_srtmode\_set:1> / \*NO / \*GROUP / \*COLLATE / \*STACKER / \*AUTOMATIC**

Stellt den Sortiermodus des Druckers ein.

**SRTMODE=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

Mit dieser Angabe legen Sie fest, dass der Sortiermechanismus für den aktuellen Auftrag benutzt wird. Der Sortiermechanismus ist zulässig für die Drucker 9026-RENO, 4818-PCL, 4822-PCL und 4825-PCL. Er kann für bis zu 20 Papierausgabefächer eingesetzt werden. Er kann nicht durch die Standardwerte kontrolliert werden, die im Geräteeintrag definiert sind.

**SRTMODE=\*NO**

Alle Seiten des Dokuments werden von unten nach oben in den Sortierfächern abgelegt. Ausnahme 9026-RENO: Hier werden die Seiten je nach optimaler Zugänglichkeit in ein Papierfach abgelegt.

**SRTMODE=\*GROUP**

Jedes Exemplar einer bestimmten einzelnen Seite - so viele bei PAGECOP angegeben wurden - wird einzeln in einem Sortierfach abgelegt. D.h. erst die in der Reihenfolge folgende Seiten wird wieder in dasselbe Sortierfach abgelegt. Nach dem Ende des Druckauftrags enthält jedes benutzte Sortierfach eine Kopie des ausgedruckten Dokuments.

*Beispiel*

Die Datei enthält drei Seiten, und es wurde PAGECOP=2 angegeben:

Seite 3	Seite 3	
Seite 2	Seite 2	
Seite 1	Seite 1	
<b>Fach 1</b>	<b>Fach 2</b>	<b>Fach 3</b>

**SRTMODE=\*COLLATE**

Alle Exemplare einer Seite - so viele bei PAGECOP angegeben wurden - werden in einem Sortierfach gesammelt. Die Exemplare der folgenden Seite werden im nächsten Sortierfach abgelegt. Die Sortierfächer werden von unten nach oben benutzt.

*Beispiel*

Die Datei enthält drei Seiten, und es wurde PAGECOP=2 angegeben:

Seite 1	Seite 2	Seite 3
Seite 1	Seite 2	Seite 3
<b>Fach 1</b>	<b>Fach 2</b>	<b>Fach 3</b>

**SRTMODE=\*STACKER**

Nicht für den Drucker 9026 anwendbar.

Alle gedruckten Seiten werden im Massenausgabefach des Sortiermechanismus' abgelegt; und zwar bis zu einem Maximum von 500 Seiten. Dieser Modus bietet sich dafür an, wenn eine einzige Kopie von einem sehr langen Dokument ausgedruckt werden soll.

**SRTMODE=\*AUTOMATIC**

Nur für den Drucker 9026 anwendbar.

Abhängig von der in PAGECOP angeforderten Anzahl der Kopien pro Seite und der Anzahl der Sortierfächer wird das Ablageverhalten automatisch gewählt. Die bedruckten Seiten werden wie bei \*NO unsortiert abgelegt, wenn die bei PAGECOP angegebene Anzahl der Kopien gleich der Anzahl der Sortierfächer ist. Die bedruckten Seiten werden wie bei \*GROUP nach Dokumenten sortiert, wenn die bei PAGECOP angegebene Anzahl der Kopien kleiner als die Anzahl der Sortierfächer ist. Oder die bedruckten Seiten werden wie bei \*COLLATE nach Seiten sortiert in den Sortierfächern abgelegt, wenn die bei PAGE-COP angegebene Anzahl der Kopien größer als die Anzahl der Sortierfächer ist.

**SRTMODE=<var: enum-of \_srtmode\_set:1>**

Die Vereinbarung bzgl. des Sortiermechanismus wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

1	NO
2	GROUP
3	COLLATE
4	STACKER
5	AUTOMATIC
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

Diese Werte sind in der ASSEMBLER-, C- und COBOL-Schnittstelle definiert. In der C-Schnittstelle steht darum z.B. Folgendes:

```

/* srtmode_set */
/* ENUM _srtmode_set */
<< Die Vereinbarung bzgl. des Sortiermechanismus wird >>
<< nicht explizit, sondern über folgende Konstantendefinition angeben>>
<< (in der Parameterliste des Funktionsaufrufs kann darum der >>
<< in der Konstanten hinterlegte ganzzahlige Wert verwendet werden): >>
#define PRNTDOCsrtmode_no 1 /* srtmode =*no */
#define PRNTDOCsrtmode_group 2 /* srtmode =*group */
#define PRNTDOCsrtmode_collate 3 /* srtmode =*collate */
#define PRNTDOCsrtmode_stack 4 /* srtmode =*stacker */
#define PRNTDOCsrtmode_automatic 5 /* srtmode =*automatic */

```

**VIRTUAL=\*UNCHANGED / <var: enum-of \_virtual\_set:1> / \*ALLOWED / \*NOT\_ALLOWED / \*STD / \*MUST**

Dieser Operand erlaubt es, den Druckauftrag über einen virtuellen Drucker an eine Anwendung zum Bearbeiten weiterzugeben.

**VIRTUAL=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**VIRTUAL=<var: enum-of \_virtual\_set:1>**

Die Vereinbarung, ob und wie ein Druckauftrag über einen virtuellen Drucker an eine Anwendung zum Bearbeiten weitergegeben wird, wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Funktion:

0	ALLOWED
1	NOT-ALLOWED
siehe Beschreibung der Parameterliste ab <a href="#">Seite 286</a> .	

Diese Werte sind in der ASSEMBLER-, C- und COBOL-Schnittstelle definiert. In der C-Schnittstelle steht darum z.B.Folgendes:

```

/* virtual_set                                     */
/* ENUM _virtual_set                               */
<< Die Vereinbarung, ob und wie ein Druckauftrag über einen virtuellen >>
<< Drucker an eine Anwendung zum Bearbeiten weitergegeben wird, wird >>
<< nicht explizit, sondern über folgende Konstantendefinition angegeben>>
<< (in der Parameterliste des Funktionsaufrufs kann darum der >>
<< in der Konstanten hinterlegte ganzzahlige Wert verwendet werden): >>
#define PRNTDOCvirtual_allowed 0                    /* virtual = *allowed      */
#define PRNTDOCvirtual_not-allowed 1                /* virtual = *not-allowed  */

```

**VIRTUAL=\*ALLOWED**

Der Druckauftrag wird bevorzugt an ein virtuelles Gerät übergeben. Sollte beim Eintragen des Druckauftrags in die Liste der Druckaufträge kein virtuelles Gerät aktiv sein, wird der Druckauftrag einem realen Gerät zugewiesen.

**VIRTUAL=\*NOT\_ALLOWED**

Dieser Druckauftrag soll nicht über ein virtuelles Gerät an eine Anwendungsprogramm weitergegeben werden.



**VIRTUAL=\*STD**

Der virtuelle Drucker, der den Druckauftrag erhalten soll, wird aus der SPOOL-Parameter-Datei ermittelt.

Der Druckauftrag wird bevorzugt an einen virtuellen Drucker übergeben. Sollte beim Eintragen des Druckauftrags in die Liste der Druckaufträge kein virtueller Drucker aktiv sein, wird der Druckauftrag einem realen Drucker zugewiesen.

**VIRTUAL=\*MUST**

Der Druckauftrag wird an einen bestimmten virtuellen Drucker übergeben, der mit den Parametern in der Klammer festgelegt wird. Sollte die Supervisor-Task dieses virtuellen Druckers inaktiv sein, bleibt der Druckauftrag im Wartezustand.

**PRJCLAS=\*UNCHANGED / <var: int: 1> / <integer 1..255>**

Nur für den SPOOL-Administrator.

Bestimmt die Auftragsklasse des SPOOL-OUT-Auftrags.

**PRJCLAS=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**PRJCLAS=<var: int: 1>**

Name eines Feldes, das mit FL definiert ist. In diesem Feld wird eine Ganzzahl (Länge 1 Byte), die als Nummer der Auftragsklasse interpretiert wird, hinterlegt.

**PRJCLAS=<integer 1..255>**

Die angegebene Auftragsklasse wird dem SPOOL-OUT-Auftrag zugewiesen. Der Wert 0 (Null) ist aus Kompatibilitätsgründen nicht zulässig.

**PROGRAM=(*name, string*)**

Legt den Gerätenamen eines virtuellen Druckers fest, von dem der SPOOL-OUT-Auftrag zu verarbeiten ist.

*name: \*UNCHANGED / <var: char: 8> /*

*<c-string 1..8: alphanum\_name\_1..8, c-string\_1..8> / \*ANY*

Legt den Gerätenamen eines virtuellen Druckers fest, von dem der SPOOL-OUT-Auftrag zu verarbeiten ist.

*name: \*UNCHANGED*

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

*name: <var: char: 8>*

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 8 Byte) hinterlegt, die als Name des virtuellen Druckers interpretiert wird.

*name: <c-string 1..8: alphanum\_name\_1..8, c-string\_1..8>*

Der Name des virtuellen Druckers, der den SPOOL-OUT-Auftrag verarbeiten soll, wird über eine in Hochkomma eingeschlossene Zeichenfolge angegeben.

*name: \*ANY*

Ein beliebiger virtueller Drucker darf den SPOOL-OUT-Auftrag verarbeiten.

*string*: **\*NONE** / <var: char: 32> /

<c-string 1..32: alphanum\_name\_1..32, c-string\_1..32>

Legt Zusatzinformationen für den virtuellen Drucker fest, von dem der SPOOLOUT-Auftrag zu verarbeiten ist.

*string*: **\*NONE**

Vorgabewert dieses Operanden. Keine Zusatzinformationen für den virtuellen Drucker.

*string*: <var: char: 32>

Name eines Feldes, das mit CL definiert ist. In diesem Feld wird eine Zeichenfolge (Länge 32 Byte) hinterlegt, die als Zusatzinformation für den virtuellen Drucker interpretiert wird.

*string*: <c-string 1..32: alphanum\_name\_1..32, c-string\_1..32>

Die Zusatzinformation für den virtuellen Drucker, der den SPOOLOUT-Auftrag verarbeiten soll, wird über eine in Hochkomma eingeschlossene Zeichenfolge angegeben.

**FRMNAME=\*UNCHANGED / \*STD / <c-string: c-string 1..63> / <var: char: 63> / (<reg: char:63>)**

Angabe des Formats, das verarbeitet werden soll. Das Dokument wird im Transparent-Modus bearbeitet.

Name des Formats, in dem die auszugebenden Daten an den Drucker übergeben werden. Wenn kein Wert für FRMNAME angegeben wurde, wird automatisch ein Vorgabewert aus dem Wert des Operanden CONTMODE abgeleitet. Dabei ergibt sich für:

CONTMODE=\*PAGE-MODE(...)

FRMNAME=\*HP

CONTMODE=\*APA(...)

FRMNAME=\*SPDS

CONT-MODE=\*LOGICAL / \*PHYSICAL / \*LINEMODE

FRMNAME=\*STD

Der Wert FRMNAME=\*STD bedeutet, dass der Inhalt der auszudruckenden Datei nicht relevant ist.

**FRMNAME=\*UNCHANGED**

Vorgabewert des Operanden, der eingestellte Wert bleibt unverändert.

**FRMNAME=\*STD**

Der Wert für FRMNAME wird auf \*STD gesetzt.

**FRMNAME=<c-string: c-string 1..63>**

Der Name des Formats wird direkt angegeben.

**FRMNAME=<var: char: 63>**

Der Name des Formats wird indirekt über ein Feld angegeben.

**FRMNAME=(<reg: char:63>)**

Ein Register zeigt auf ein Feld, das den Namen des Formats enthält.

**TOPOFF=\*UNCHANGED / \*IGNORE / <integer -255..255>**

Legt den Abstand des oberen Randes der Druckseite vom oberen Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der Druckseite nicht beachtet wird.

**TOPOFF=\*UNCHANGED**

Vorgabewert des Operanden, der eingestellte Wert bleibt unverändert.

**TOPOFF=\*IGNORE**

Die Druckseite wird nicht in senkrechter Richtung auf dem Papier verschoben.

**TOPOFF=<integer -255..255>**

Die Druckseite wird um den angegebenen Weg in senkrechter Richtung auf dem Papier verschoben.

**LEFTOFF=\*UNCHANGED / \*IGNORE / <integer -255..255>**

Legt den Abstand des linken Randes der Druckseite vom linken Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der Druckseite nicht beachtet wird.

**LEFTOFF=\*UNCHANGED**

Vorgabewert des Operanden, der eingestellte Wert bleibt unverändert.

**LEFTOFF=\*IGNORE**

Die Druckseite wird nicht in waagerechter Richtung auf dem Papier verschoben.

**LEFTOFF=<integer -255..255>**

Die Druckseite wird um den angegebenen Weg in waagerechter Richtung auf dem Papier verschoben.

**SCHEDTIME=(*schdate*, *schttime*)**

Bestimmt die Scheduling-Zeit, d.h. die Zeit ab der ein Druckauftrag frühestens gestartet wird. Damit wird der Druckauftrag erst ab dem angegebenen Zeitpunkt entsprechend der Druckerverfügbarkeit und den Abarbeitungsregeln für Druckaufträge gestartet. Im Fall, dass sich Aufträge im WFT-Status befinden, kann der Datentransfer ab diesem Moment ausgelöst werden. Es ist nicht erlaubt, diesen Operanden in Interoperabilität mit UNIX-Systemen zu bringen.

*schdate*: **\*UNCHANGED** / **\*TODAY** / <var: char 1..10> / <c-string 10..10>

Bestimmt den Tag, an dem ein Druckauftrag gestartet wird.

*schdate*: **\*UNCHANGED**

Vorgabewert des Operanden, der eingestellte Wert bleibt unverändert.

***schdate:* \*TODAY**

Der Tag ist der gegenwärtige Tag.

***schdate:* <var: char 1..10> / <c-string 10..10>**

Bestimmt das Datum in der Form <Jahr>-<Monat>-<Tag>. Diese Angabe setzt sich aus drei unsigned Integerwerten zusammen, die durch Bindestriche getrennt werden. Die Jahresangabe hat vier Stellen. Die Monatsangabe hat bis zu zwei Stellen mit den möglichen Zahlen 1..12. Die Tagesangabe hat auch bis zu zwei Stellen mit den möglichen Zahlen 1..31, wobei die Tagesangabe mit der Monatsangabe stimmig sein muss. D.h. es darf z.B. nicht 2000-02-30 angegeben werden.

***schtime:* \*UNCHANGED / \*NOW / <var: char 1..5> / <c-string 5..5>**

Bestimmt den Zeitpunkt, an dem ein Druckauftrag gestartet wird.

***schtime:* \*UNCHANGED**

Vorgabewert des Operanden, der eingestellte Wert bleibt unverändert.

***schtime:* \*NOW**

Der Zeitpunkt ist der gegenwärtige Zeitpunkt.

***schtime:* <var: char 1..5> / <c-string 5..5>**

Bestimmt den Zeitpunkt des Tages in der Form <Stunde>:<Minute>. Diese Angabe setzt sich aus zwei unsigned Integerwerten mit jeweils 2 Stellen zusammen, die durch einen Doppelpunkt getrennt sind. Die möglichen Zahlen sind 0..23 bei den Stunden und 0..59 bei den Minuten. Bei einstelligen Werten kann eine Null vorangestellt werden.

### *Beispiel*

SCHEDTIME = (SCHDATE=\*TODAY, SCHTIME=\*NOW)

Es gibt keine Scheduling-Zeit. Der Druckauftrag wird entsprechend der Druckerverfügbarkeit und den Abarbeitungsregeln für Druckaufträge gestartet.

### **Unverträgliche Operanden**

Wenn der Operand VIRTUAL auf \*NOT\_ALLOWED gesetzt ist, darf der Operand PROGRAM nicht mit einem Wert belegt sein, sonst wird ein Fehler gemeldet.

**Returncodes**

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	Verarbeitung fehlerfrei beendet
00	00	0048	Warnung: DELETE (Löschen) für Systemdatei ignoriert
00	01	0001	ungültiger Dateiname
00	01	0002	ungültiger PL-Zugriff
00	01	0008	Inkonsistente Parameter
00	01	0024	ungültige Parameterangaben
00	01	0030	RECFORM=F zusammen mit RECSIZE=0 unzulässig
00	01	0078	Drucken von *SYSOUTPUT unzulässig
00	01	007C	l-spac=ebcdic und RECFORM=U
00	01	0080	Drucken von *SYSOUT unzulässig
00	01	0090	ungültiger oder fehlender Parameter für HP-HP90
00	01	0098	Systemdatei zu *DUMMY zugewiesen
00	01	009C	Systemdatei zu *TAPE zugewiesen
00	01	00A0	Systemdatei zu temporärer Datei zugewiesen
00	01	00A4	Systemdatei nicht zu SAM-Datei zugewiesen
00	01	00DC	keine PLAM-Bibliothek
00	01	FFFF	Interface-Version nicht unterstützt
00	20	001C	Fehler bei Speicheranforderung
00	20	0020	Fehler bei Speicherfreigabe
00	20	0004	Fehler der Warteschlangenroutine
00	20	00D4	keine Information über den Benutzerauftrag
00	20	00FC	interner Fehler
00	40	000C	DVS-Fehler
00	40	0010	Fehler in EAM-Verarbeitung
00	40	0018	ungültiges oder fehlendes Kennwort
00	40	002C	ungültiger Dateizugriff
00	40	0034	Erzeugungs-/Verfallsdatum nicht verfügbar
00	40	0040	EAM-Datei nicht vorhanden
00	40	0044	Systemdatei enthält keine Daten
00	40	0054	Kommando DELETE-FILE aktiv
00	40	006C	Fehler beim Schließen einer Systemdatei
00	40	0070	Datei gerade im Gebrauch
00	40	0088	Ausgabeziel existiert nicht
01	40	00A8	Dateizugriff nicht verfügbar
02	40	00b4	Formular existiert nicht
03	40	00B8	SAM-Fehler in EAM-Verarbeitung
04	40	00BC	Rotation nicht unterstützt
05	40	00C0	CHAR-SET-NUM oder FOB-Größe überschritten

(SC2)	SC1	Maincode	Bedeutung
06	40	00C4	CHAR-SET-NUM im Pool überschritten
07	40	00C8	Kein ROTATION-LOOP definiert
08	40	00CC	neue TSN kann nicht zugewiesen werden
09	40	00E0	PLAM-Bibliothek nicht vorhanden
04	40	00E4	Bibliothek nicht gefunden
05	40	00E8	MONJV-Fehler
06	40	00EC	Authorisierungsfehler
07	40	00F0	verteilte Verarbeitung unzulässig
08	80	0094	SPOOL Subsystem angehalten
09	80	FFFF	S&P Subsysteme nicht geladen

### Hinweise

- Das Makro MODPJAT gibt obige Returncodes zurück. Zusätzlich können die Returncodes von HDRCHECK und \$VALID zurückgegeben werden. Wenn kein Datenbereich für den Header bereitgestellt wurde, wird der Wert X'0000003C' im Register 15 zurückgegeben.
- Das Makro MODPJAT gibt zu den folgenden Subsystemen spezifische Informationen zusammen mit der Information aus, ob das Subsystem geladen ist:
  - RSO
  - Dprint
  - SPS

### Beschreibung der Parameterliste

```
* parameter list description
SMAPL DS OF
SMAHDR FHDR MF=(C,SMA),EQUATES=NO Standard header
* main return codes
SMAOK EQU 0 successfull processing
SMAW048 EQU 72 warning
SMAE001 EQU 1 invalid filename
SMAE002 EQU 2 invalid pl access
SMAE008 EQU 8 inconsistency between param
SMAE024 EQU 36 invalid parameter spec
SMAE030 EQU 48 recform=F not allowed with
* reccsize=0
SMAE078 EQU 120 print *sysout not allowed
SMAE07C EQU 124 l-spac=ebcdic and recform U
SMAE080 EQU 128 fgg not allowed
SMAE090 EQU 144 invalid or missing parameters
* for HP-HP90
SMAE098 EQU 152 system file assigned to
```

*			*dummy
SMAE09C	EQU	156	system file assigned to tape
SMAE0A0	EQU	160	system file assigned to
*			temporary one
SMAE0A4	EQU	164	system file not assigned to
*			SAM file
SMAE0DC	EQU	220	not a plam library
SMAWVER	EQU	65535	interface version not
*			supported
SMAE01C	EQU	28	memory request error
SMAE020	EQU	32	memory release error
SMAE004	EQU	4	queue routine error
SMAE0D4	EQU	212	no information on user job
SMAE0FC	EQU	252	internal error
SMAE00C	EQU	12	DMS error
SMAE010	EQU	16	error in EAM processing
SMAE018	EQU	24	invalid or missing password
SMAE02C	EQU	44	invalid file access
SMAE034	EQU	52	creation-expiration date not
*			available
SMAE040	EQU	64	EAM file not available
SMAE044	EQU	68	syst file contains no data
SMAE054	EQU	84	DELETE-FILE command active
SMAE06C	EQU	108	syst file close error
SMAE070	EQU	112	file currently in use
SMAE088	EQU	136	destination does not exist
SMAE0A8	EQU	168	file access not available
SMAE0B4	EQU	180	form does not exist
SMAE0B8	EQU	184	SAM error in EAM processing
SMAE0BC	EQU	188	no rotation supported
SMAE0C0	EQU	192	exceeding char-set-num or FOB
*			size
SMAE0C4	EQU	196	exceeding char-set-num in
*			pool
SMAE0C8	EQU	200	no rotation-loop defined
SMAE0CC	EQU	204	new TSN can not be assigned
SMAE0E0	EQU	224	PLAM library not available
SMAE0E4	EQU	228	library element not found
SMAE0E8	EQU	232	monjv error
SMAE0EC	EQU	236	authorization error
SMAE0F0	EQU	240	distribution not allowed
SMAE094	EQU	148	SPOOL subsystem pended
SMAFFFF	EQU	65535	S&P subsystems not available
*			
SMARSOP	DS	A	address of rso_pl
SMASPP	DS	A	address of sps_pl
SMADPP	DS	A	address of dprint_pl
SMASEMP	DS	A	address of dsem_pl

SMARES1	DS	A	
SMARES2	DS	A	
SMARES3	DS	A	
SMARES4	DS	A	
SMATSN	DS	CL4	client or server TSN
SMACLUS	DS	CL8	cluster name
SMASERV	DS	CL8	server name
SMAMJV	DS	CL54	MONJV name
SMAUN1	DS	XL14	
SMAMOPT	DS	FL1	modify option
* _option_set			
SMAOPTS	EQU	1	option = tsn
SMAOPSR	EQU	2	option = server-tsn
SMAOPCL	EQU	3	option = client-tsn
SMAOPMO	EQU	4	option = monjv
*			
SMAUN2	DS	XL147	
*			
SMASECR	DS	0XL64	input-section
SMASEID	DS	CL60	section-identifier
SMASEPO	DS	H	position
SMASELE	DS	X	section-id length
SMASEIN	DS	AL1	sect_ind
SMASEGI	EQU	X'80'	sect-id given
SMASEIC	EQU	X'40'	character = true or
*			
hexadecimal = false			
SMASEWF	EQU	X'20'	sect-id = *whole-file
SMASEPS	EQU	X'10'	position = *std
SMASEST	EQU	X'08'	on : sect-id is given in
*			
char, off : sect-id is given			
*			
in hexa			
SMASETG	EQU	X'04'	section type has been given
SMASERF	EQU	X'03'	
*			
*			
SMAFREC	DS	0XL72	input-part(first-record)
SMASPOS	DS	F	record number
SMASSID	DS	CL60	string
SMASSPO	DS	H	string position
SMASROC	DS	H	string occurrence
SMASLE	DS	X	string length
SMAFIL6	DS	CL1	
SMASII	DS	Y	strid_ind
SMASSIC	EQU	X'8000'	character = true or
*			
hexadecimal = false			
SMASB0	EQU	X'4000'	first-rec = *begin-of-file
SMASIT	EQU	X'2000'	first-rec = integer
SMASBS	EQU	X'1000'	first-rec = *by-string



SMASSPS	EQU	X'0800'	position = *std
SMASST	EQU	X'0400'	on : first-rec given in char,
*			off : first-rec is given in
*			hexa
SMASSTG	EQU	X'0200'	section type has been given
SMASSIF	EQU	X'01FF'	
*			
*			
SMALREC	DS	0XL72	input-part(last-record)
SMAEPOS	DS	F	record number
SMAESID	DS	CL60	string
SMAESPO	DS	H	string position
SMAETOC	DS	H	string occurrence
SMAESLE	DS	X	string length
SMAFIL7	DS	CL1	
SMAESII	DS	Y	strid_ind
SMAEEIC	EQU	X'8000'	character = true or
*			hexadecimal = false
SMAESEF	EQU	X'4000'	last-rec = *end-of-file
SMAESIT	EQU	X'2000'	last-rec = integer
SMAESBS	EQU	X'1000'	first-rec = *by-string
SMAESPS	EQU	X'0800'	position = *std
SMAESST	EQU	X'0400'	on : last-rec given in char,
*			off : last-rec is given in
*			hexa
SMAESTG	EQU	X'0200'	section type has been given
SMAESIF	EQU	X'01FF'	
*			
*			
SMARECP	DS	0XL4	record-part
SMARPC	DS	H	first character
SMARPLC	DS	H	last character
*			
*			
SMALISP	DS	0XL4	line-spacing
SMACCP0	DS	H	control char position
SMASPAC	DS	FL1	line spacing
*			
* _space_set			
SMASPA1	EQU	1	spacing = *space_1
SMASPA2	EQU	2	spacing = *space_2
SMASPA3	EQU	4	spacing = *space_3
SMASPAE	EQU	8	spacing = *by_ebcdic_control
SMASPAA	EQU	16	spacing = *by_asa_control
SMASPAI	EQU	32	spacing = *by_ibm_control
SMASPA	EQU	144	spacing = *no
SMASPAS	EQU	0	spacing = *std
*			
SMALSUN	DS	XL1	unused

```

*
SMADOCF DS FL1 document format
* _docform_set
SMADCFT EQU 0 docform = *text
SMADCFP EQU 1 docform = *page-format
SMADCFE EQU 2 docform = *special-format
*
*
SMACONM DS OXL3 control mode
SMACTLM DS FL1 control mode
* _control_mode_set
SMACTNO EQU 0 mode = *none
SMACTMS EQU 1 mode = *std
SMACTPH EQU 2 mode = *physical
SMACTMP EQU 4 mode = *page-mode
SMACTML EQU 8 mode = *logical
SMACTLI EQU 16 mode = *line-mode
SMACTMA EQU 64 mode = *apa
*
SMAPCC DS FL1 page control character
* _pcc_set
SMATOPY EQU 0 pcc = *yes
SMATOPN EQU 1 pcc = *no
SMATOPD EQU 2 pcc = *dummy
*
SMACTLT DS FL1 control type
* _control_type_set
SMACTDU EQU 0 ctltype = *dummy
SMACTHC EQU 1 ctltype = *compatible
SMACTHP EQU 2 ctltype = *hp
*
*
SMAUN4 DS XL1
SMACHKP DS FL1 checkpoint
* _checkp_set
SMACKPP EQU 0 checkp = *on-page
SMACKPR EQU 1 checkp = *on-section-record
*
SMADOCN DS H Number of files
SMAUN5 DS XL5
SMALOCK DS FL1 lock file
* _lockf_set
SMALOCS EQU 0 lockf = *std
SMALOCN EQU 1 lockf = *no
SMALOCY EQU 2 lockf = *yes
*
SMADELFF DS FL1 delete-after-print
* _delf_set

```

SMADELY	EQU	1	delf = *yes
SMADELD	EQU	2	delf = *destroy
SMADELN	EQU	3	delf = *no
*			
SMACOPY	DS	X	additional-copies
SMAUN6	DS	CL58	
SMAPRJN	DS	CL8	print-job-name
SMAPRIO	DS	X	print-job-priority
SMAUN7	DS	XL24	
SMATRNC	DS	FL1	Line-truncation
*		_truncation_set	
SMATRUS	EQU	0	trunc = *std
SMATRUD	EQU	1	trunc = *delete_file
SMATRUK	EQU	2	trunc = *keep_file
*			
*			
SMAPART	DS	0XL12	output-part
SMARCFR	DS	F	from
SMARCTO	DS	F	to
SMALAYD	DS	FL1	dimension
*		_layout_part_dim_set	
SMALAYN	EQU	0	dim = *none
SMALAYP	EQU	1	dim = *pages
SMALAYL	EQU	2	dim = *line
*			
SMAOPSW	DS	AL1	sect_ind
SMAOIFB	EQU	X'80'	sect-id given
SMAOITE	EQU	X'40'	character = true or
*			hexadecimal = false
SMAOIFR	EQU	X'3F'	
SMAFIL5	DS	CL2	
*			
SMACLAS	DS	X	print-job-class
SMALFTM	DS	X	left-margin
SMALIPP	DS	H	Lines-per-page
SMAHDLI	DS	FL1	header-line
*		_headline_set	
SMAHLNO	EQU	1	headlin = *no
SMAHLST	EQU	2	headlin = *std
SMAHLR	EQU	4	headlin = *frecord
SMAHLD	EQU	8	headlin = *date
SMAHLDR	EQU	12	headlin = *date_frecord
SMAHLP	EQU	16	headlin = *page
SMAHLRP	EQU	20	headlin = *frecord_page
SMAHLDP	EQU	24	headlin = *date_page
SMAHDRP	EQU	28	headlin = *date_frecord_page
*			
SMAOUTF	DS	FL1	output-format

```

*   _outform_set
SMAOUTC EQU 1           outform = *character
SMAOUTH EQU 2           outform = *hexadecimal
*
SMATWSD DS FL1           two-sided
*   _twoside_set
SMATWOS EQU 0           twoside = *std
SMATWOY EQU 1           twoside = *yes
SMATWOT EQU 2           twoside = *tumble
SMATWON EQU 4           twoside = *no
SMATWOI EQU 8           twoside = *ignore
*
SMAROT DS FL1           rotation
*   _rotation_set
SMAROTN EQU 213         rot = *no
SMAROT0 EQU 240         rot = *rot_0
SMAROT1 EQU 241         rot = *rot_90
SMAROT2 EQU 242         rot = *rot_180
SMAROT3 EQU 243         rot = *rot_270
SMAROT4 EQU 244         rot = *rot_0_180
SMAROT5 EQU 245         rot = *rot_90_270
SMAROT6 EQU 246         rot = *rot_180_0
SMAROT7 EQU 247         rot = *rot_270_90
SMAROTY EQU 232         rot = *by_control_codes
SMAROTD EQU 255         rot = *no for modify
*
SMAIT# DS OXL1          intray
SMAINT DS X             input-tray-number
          ORG SMAIT#
SMAITS DS FL1          input-tray indicator
*   _intray_set
SMAITBF EQU 0           intray = *by-format
SMAITIG EQU 255         intray = *ignore
*
          ORG SMAIT#
SMAITFS DS FL1          input-tray-format
*   _intray_format_set
SMAITMA EQU 128         intray_format = *manual
SMAITA3 EQU 129         intray_format = *a3
SMAITA4 EQU 130         intray_format = *a4
SMAITA5 EQU 131         intray_format = *a5
SMAITB4 EQU 132         intray_format = *b4
SMAITB5 EQU 133         intray_format = *b5
SMAITDB EQU 134         intray_format =
*                        *double-letter
SMAITEX EQU 135         intray_format = *exec
SMAITF0 EQU 136         intray_format = *folio
SMAITIV EQU 137         intray_format = *invoice

```

SMAITLG	EQU	138	inray_format = *legal
SMAITLT	EQU	139	inray_format = *letter
SMAITMO	EQU	140	inray_format = *monarch
SMAIT10	EQU	141	inray_format =
*			*commercial-10
SMAITDL	EQU	142	inray_format = *d1
SMAITC5	EQU	143	inray_format = *c5
SMAIT3U	EQU	144	inray_format = *A3-uncut
SMAIT4U	EQU	145	inray_format = *A4-uncut
SMAITLD	EQU	146	inray_format = *ledger
*			
	ORG	SMAIT#+1	
SMAOT#	DS	0XL1	outtray
SMAOUTT	DS	X	output-tray-number )
	ORG	SMAOT#	
SMAOTS	DS	FL1	output-tray indicator
*			_outtray_set
SMAOUTI	EQU	128	outtray = *ignore
SMAOUTS	EQU	129	outtray = *sorter
*			
	ORG	SMAOT#+1	
SMAPGCP	DS	H	page-copies
*			
SMACVPG	DS	0XL40	cover-pages
SMAHTXT	DS	CL32	header page text
SMAHDEX	DS	F	header exit number
SMATREX	DS	F	trailer exit number
*			
SMAFORM	DS	CL6	form-name
SMALoop	DS	CL3	loop-name
SMARTLP	DS	CL3	rotation-loop name
SMACHS	DS	16CL3	character-sets
SMACHS#	EQU	16	
SMAPLNM	DS	CL4	pool-name
SMAPLIN	DS	X	pool-index
SMAEFO	DS	CL2	electronic-overlay
*			
SMAOVLY	DS	0XL2	overlay
SMAOVFA	DS	X	face side
SMAOVRE	DS	X	reverse side
*			
SMAFOB	DS	CL4	forms-overlay-buffer
SMAVIRT	DS	FL1	virtual-printer
*			_virtual_set
SMAMUST	EQU	4	virtual = *must
SMAALLO	EQU	2	virtual = *allowed
SMANOTA	EQU	1	virtual = *not_allowed
SMASTDA	EQU	0	virtual = *std

```

*
SMAFIL2 DS CL2
SMAPGPC DS F page-definition
SMAUSRR DS CL44 user-resources-file
*
SMATRTB DS 0XL52 translation-table
SMATRNA DS CL8 translation table name
SMATRFI DS CL44 translation table file
*
SMAPRNM DS CL8 printer-name
SMAPRTY DS FL1 printer-type
* _prtype_set
SMAPTAN EQU 0 prtype = *any
SMAPT65 EQU 1 prtype = *lp65_printer
SMAPTAP EQU 2 prtype = *apa_printer
SMAPTHP EQU 3 prtype = *hp_printer
*
SMASRTM DS FL1 sorter mode
* _srtmode_set
SMASMNO EQU 1 srtmode = *no
SMASMGR EQU 2 srtmode = *group
SMASMCL EQU 3 srtmode = *collate
SMASMST EQU 4 srtmode = *stacker
SMASMAU EQU 5 srtmode = *automatic
*
SMAFIL3 DS CL2
SMATSNO DS CL4 OUTPUT : returned tsn of the
* modified job
SMAFIL4 DS CL8
SMASP1 DS AL1 specified1
SMASP11 EQU X'F0'
SMASP15 EQU X'08' section record
SMASP16 EQU X'04' first record
SMASP17 EQU X'02' last record
SMASP18 EQU X'01' record part
SMASP2 DS AL1 specified2
SMASP21 EQU X'80' line spacing
SMASP22 EQU X'40' document format
SMASP23 EQU X'20' control mode
SMASP24 EQU X'10' additional copies
SMASP25 EQU X'08' family
SMASP26 EQU X'04' checkpoint
SMASP27 EQU X'02' monjv
SMASP28 EQU X'01' jv-password
SMASP3 DS AL1 specified3
SMASP31 EQU X'80' print-job-name
SMASP32 EQU X'40' print-job-priority
SMASP33 EQU X'20' processing admission

```

SMASP34	EQU	X'10'	line-truncation
SMASP35	EQU	X'08'	lock-file
SMASP36	EQU	X'04'	delete-after-print
SMASP37	EQU	X'02'	output part
SMASP38	EQU	X'01'	page-copies
SMASP4	DS	AL1	specified4
SMASP41	EQU	X'80'	left-margin
SMASP42	EQU	X'40'	line-per-page
SMASP43	EQU	X'20'	header-line
SMASP44	EQU	X'10'	output-format
SMASP45	EQU	X'08'	two-sided
SMASP46	EQU	X'04'	rotation
SMASP47	EQU	X'02'	input-tray
SMASP48	EQU	X'01'	output-tray
SMASP5	DS	AL1	specified5
SMASP51	EQU	X'80'	cover-page
SMASP52	EQU	X'40'	form-name
SMASP53	EQU	X'20'	loop-name
SMASP54	EQU	X'10'	rotation-loop-name
SMASP55	EQU	X'08'	character-set
SMASP56	EQU	X'04'	pool-name
SMASP57	EQU	X'02'	pool-index
SMASP58	EQU	X'01'	electronic-form-overlay
SMASP6	DS	AL1	specified6
SMASP61	EQU	X'80'	overlay
SMASP62	EQU	X'40'	fob
SMASP63	EQU	X'20'	lp65-pagedef
SMASP64	EQU	X'10'	user resources file
SMASP65	EQU	X'08'	translation table
SMASP66	EQU	X'04'	printer-name
SMASP67	EQU	X'02'	printer-type
SMASP68	EQU	X'01'	tsn
SMASP7	DS	AL1	specified7
SMASP71	EQU	X'80'	rso parameters
SMASP72	EQU	X'40'	sps parameters
SMASP73	EQU	X'20'	dprint parameters
SMASP74	EQU	X'10'	dsem parameters
SMASP7F	EQU	X'0F'	
SMASP8	DS	AL1	specified8
SMASP81	EQU	X'80'	reserved for start-processing
SMASP82	EQU	X'40'	sorter mode
SMASP83	EQU	X'20'	virtual control
SMASP84	EQU	X'10'	job-class
SMASP85	EQU	X'08'	prog. name
SMASP86	EQU	X'04'	string field
SMASP87	EQU	X'02'	format name
SMASP88	EQU	X'01'	scheduling time

\*

SMAVALL	DS	0XL40	virtual device synchrone
SMAPVNA	DS	CL8	virtual program name
SMAPVST	DS	CL32	virtual program string
*			
SMAFRMT	DS	CL63	format name
SMASPM1	DS	AL1	specified_m1
SMASPM11	EQU	X'80'	header exit number
SMASPM12	EQU	X'40'	trailer exit number
SMASPM13	EQU	X'20'	
SMASPM14	EQU	X'1F'	
SMASPM2	DS	AL1	specified_m2
SMASPM24	EQU	X'FF'	
SMASP9	DS	AL1	specified9
SMASP91	EQU	X'80'	top-offset
SMASP92	EQU	X'40'	left-offset
SMASP93	EQU	X'20'	scheduling time
SMASP9F	EQU	X'1F'	
SMATOP0	DS	H	Specifies the length of the
*			white band that must be
*			inserted
SMALEF0	DS	H	Specifies the width of the
*			white band that must be
*			inserted
SMAUN9	DS	XL11	unused
SMADATE	CL	10	specifies the date from which
			the job can be started
SMATIME	CL	5	specifies the time from which
			the job can be started
SMA#	EQU	*-SMAHDR	



## SHOPJAT - Attribute eines Druckauftrags anzeigen

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** C, D, E, L, M

Mit diesem Makro werden die Attribute eines Druckauftrags ausgegeben. Das Makro entspricht dem Kommando SHOW-PRINT-JOB-ATTRIBUTES.

### Format

Operation	Operanden
SHOPJAT	XPAND=PARAM / OUTPUT  VARIANT= <u>001</u> / <c-string 3..3>  CALLER= <u>*USER</u> / *SYSTEM  TSN=( <i>tsn1</i> , <i>cluste1</i> ) <i>tsn1</i> : <var: char: 4> / <c-string 4..4> <i>cluste1</i> : * <u>LOCAL_CLUSTER</u> / <var: char: 8> / <c-string 1..8: name_1..8>  SERVTSN=( <i>tsn2</i> , <i>server</i> ) <i>tsn2</i> : <var: char: 4> / <c-string 4..4> <i>server</i> : <var: char: 8> / <c-string 1..8: name_1..8>  MONJV= <u>*NONE</u> / <var: char: 54> / <c-string 1..54: filename_1..54_with-catid>  FOREIGN=( <i>identif</i> , <i>cluste2</i> ) <i>identif</i> : * <u>NONE</u> / <var: int: 4> / <integer 1..2147483647> <i>cluste2</i> : <var: char: 8> / <c-string 1..8: name_1..8>  BUFFER=<var: pointer>  BUFLLEN=<var: int: 4> / <integer 1..8388607>  INFO=list-poss(7): * <u>ALL</u> / * <u>PRINTER_SELECTION</u> / * <u>DOCUMENT_DESCRIPTION</u> / * <u>PRINT_JOB_CONTROL</u> / * <u>LAYOUT_CONTROL</u> / * <u>RESOURCES_DESCRIPTION</u> / * <u>TO_PRINTER</u>

## Operandenbeschreibung

### **XPAND=PARAM / OUTPUT**

Es wird festgelegt, welche Struktur zu expandieren (erzeugen) ist. Angaben bei diesem Operanden werden bei anderen MF-Werten ignoriert.

### **XPAND=PARAM**

Das Layout der Eingabe-Parameterliste wird expandiert

### **XPAND=OUTPUT**

Es werden alle Datenbeschreibungen (DSECTs) zur Beschreibung der Ausgabe-Informationenblöcke generiert.

### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **CALLER=\*USER / \*SYSTEM**

Aufrufer des Makros.

### **CALLER=\*USER**

Vorgabewert für diesen Operanden. Ein Benutzer ruft das Makro auf.

### **CALLER=\*SYSTEM**

Das Makro wird vom System aufgerufen.

### *Hinweis*

Die Operanden TSN, SERVTSN, MONJV und FOREIGN schließen sich gegenseitig aus. Daher darf von diesen Operanden immer nur einer verwendet werden.

### **TSN=(tsn1, cluste1)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Client-Rechner.

#### **tsn1: <var: char: 4> / <c-string 4..4>**

TSN des Druckauftrags, der abgebrochen werden soll.

#### **tsn1: <var: char: 4>**

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

#### **tsn1: <c-string 4..4>**

Die TSN des Druckauftrags wird direkt angegeben.

#### **cluste1: \*LOCAL\_CLUSTER / <var: char: 8> / <c-string 1..8: name\_1..8>**

Name des Clusters, in dem der Druckauftrag abgearbeitet werden sollte. Wenn ein Name angegeben wurde, bezieht sich die TSN auf den Gateway-Host im angegebenen Cluster. Es können nur BS2000-Cluster angegeben werden.

***cluste1*: \*LOCAL CLUSTER**

Vorgabewert für den Namen des Clusters, in dem der Druckauftrag abgearbeitet werden sollte.

***cluste1*: <var: char: 8>**

Der Name des Clusters wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***cluste1*: <c-string 1..8: name\_1..8>**

Der Name des Clusters wird direkt angegeben.

**SERVTSN=(*tsn2*, *server*)**

Identifiziert den Druckauftrag durch seine Referenzen auf dem Server-Rechner. Nur Druckaufträge im lokalen Cluster können auf diese Weise angesprochen werden.

***tsn2*: <var: char: 4> / <c-string 4..4: alphanum-name\_4..4>**

TSN auf dem Server-Rechner des Druckauftrags, der abgebrochen werden soll.

***tsn2*: <var: char: 4>**

Die TSN des Druckauftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***tsn2*: <c-string 4..4: alphanum-name\_4..4>**

Die TSN des Druckauftrags wird direkt angegeben.

***server*: <var: char: 8> / <c-string 1..8: name\_1..8>**

Name des Server-Rechners, auf dem der Druckauftrag abgearbeitet wird.

***server*: <var: char: 8>**

Der Name des Server-Rechners wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

***server*: <c-string 1..8: name\_1..8>**

Der Name des Server-Rechners wird direkt angegeben.

**MONJV=\*NONE / <var: char: 54> / <c-string1..54: filename\_1..54\_with-catid>**

Der Druckauftrag wird über die zu diesem Auftrag gehörende MONJV angesprochen, vorausgesetzt, dass auf die angegebene MONJV auf dem lokalen Rechner zugegriffen werden kann.

**MONJV=\*NONE**

Vorgabewert für den Operanden MONJV. Es wurde keine MONJV angegeben.

**MONJV=<var: char: 54>**

Der Name der MONJV wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**MONJV=<c-string1..54: filename\_1..54\_with-catid>**

Der Name der MONJV wird direkt angegeben.

**FOREIGN=(*identif*, *cluste2*):**

Der Druckauftrag wird über eine Nicht-BS2000-Identifikation angesprochen.

*identif*: **\*NONE** / **<var: int: 4>** / **<integer 1..2147483647>**

Identifikation des Nicht-BS2000-Verweises auf den Druckauftrag.

*identif*: **\*NONE**

Vorgabewert, keine Identifikation angegeben.

*identif*: **<var: int: 4>**

Die Identifikation wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*identif*: **<integer 1..2147483647>**

Die Identifikation wird direkt angegeben.

*cluste2*: **<var: char: 8>** / **<c-string 1..8: name\_1..8>**

Name des Clusters, in dem der Druckauftrag abgearbeitet wird.

*cluste2*: **<var: char: 8>**

Der Name des Clusters wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

*cluste2*: **<c-string 1..8: name\_1..8>**

Der Name des Clusters wird direkt angegeben.

**BUFFER=<var: pointer>**

Zeiger auf den Ausgabepuffer.

**BUFLEN=<var: int: 4> / <integer 1..8388607>**

Größe des Ausgabepuffers.

**BUFLEN=<var: int: 4>**

Die Größe des Ausgabepuffers wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**BUFLEN=<var: int: 4> / <integer 1..8388607>**

Die Größe des Ausgabepuffers wird direkt angegeben.

**INFO=list-poss(7): \*ALL / \*PRINTER\_SELECTION / \*DOCUMENT\_DESCRIPTION / \*PRINT\_JOB\_CONTROL / \*LAYOUT\_CONTROL / \*RESOURCES\_DESCRIPTION / \*TO\_PRINTER**

Mit diesem Operanden kann ausgewählt werden, welche Informationen ausgegeben werden sollen.

**Returncodes**

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	kein Fehler
02	00	0000	Druckauftrag nicht gefunden oder Verarbeitung nicht erlaubt
01	01	FFFF	Ungültige Parameter
02	01	FFFF	ungültige Adresse / Länge
03	01	FFFF	Ausgabebereich zu kurz
00	20	FFFF	Systemfehler
02	40	FFFF	keine Autorisation
04	40	FFFF	Wertefehler im Operanden
05	40	FFFF	P/L-Version nicht unterstützt
00	80	FFFF	Subsystem nicht geladen
01	80	FFFF	SPOOL nicht geladen
02	80	FFFF	RSO nicht geladen
03	80	FFFF	PLAM nicht geladen
04	80	FFFF	JV nicht geladen
05	80	FFFF	SPS nicht geladen
06	80	FFFF	DPRINTCL nicht geladen
07	80	FFFF	DPRINTCM nicht geladen
08	80	FFFF	DPRINTSV nicht geladen
09	80	FFFF	DSEM nicht geladen

*Hinweise*

- Das Makro SHOPJAT gibt den Returncode zurück. Zusätzlich können die Returncodes von HDRCHECK und \$VALID zurückgegeben werden.  
Wenn kein Datenbereich für den Header bereitgestellt wurde, wird der Wert X'0000003C im Register 15 zurückgegeben.
- Die Information über den Druckauftrag wird in einem Ausgabepuffer abgelegt, der vom Benutzer bereitgestellt werden muss. Wenn der Puffer zu kurz definiert ist, wird er über die vorhandene Länge gefüllt. Die Abschnitte `_main_header_model` und `_header_block_mdl` werden immer in den Puffer geschrieben. Sofern sie nicht 0 enthalten, enthalten die jeweils zwei Byte langen Felder im Abschnitt `_main_header_model` den Versatz vom Beginn des Ausgabepuffers zum dazugehörigen Datenblock.

Benutzte Länge	Gesamtanzahl Druckaufträge	gemeldete Zahl Druckaufträge	Gesamtanzahl PAM-Seiten	Adresse erster Block
Blocklänge	Block 1			
Blocklänge	Block 2			
Blocklänge	Block 3			

- Das Makro SHOPJAT gibt zu den folgenden Subsystemen spezifische Informationen zusammen mit der Information aus, ob das Subsystem geladen ist:
  - RSO
  - Dprint
  - SPS

### Eingabedatenbeschreibung für die Programmiersprache Assembler

```

* parameter list description
SJA_PL_MDL          DS    0F
SJAHDR  FHDR  MF=(C,SJA),EQUATES=NO    Standard header
* main return codes
SJARCNO            EQU    0              subcode1 = 00 : no error
SJAWARN            EQU    2              job not found or processing
* not permitted
SJAPARM            EQU    1              subcode1 = 01 : wrong
* parameter
SJAVALD            EQU    2              invalid address/length
SJASHRT            EQU    3              output area too short
SJASYSE            EQU    0              subcode1 = 20 : system error
SJAPRIV            EQU    2              subcode1 = 40 : no
* authorization
SJAVERR            EQU    4              operand value error
SJAVERE            EQU    5              p/l version not supported
SJANOSS            EQU    0              subcode1 = 80 : subsystem not
* loaded

```

SJANSPO	EQU	1	spool not loaded
SJANRSO	EQU	2	rso not loaded
SJANPLM	EQU	3	plam not loaded
SJANJV	EQU	4	ju not loaded
SJANSPS	EQU	5	sps not loaded
SJANDCL	EQU	6	dprintcl not loaded
SJANDCM	EQU	7	dprintcm not loaded
SJANDSV	EQU	8	dprintsv not loaded
SJANDSM	EQU	9	dsem not loaded
*			
SJATSN	DS	CL4	client or server TSN
SJACLUS	DS	CL8	cluster name
SJASERV	DS	CL8	server name
SJAMJV	DS	CL54	monju name
SJAUN2	DS	XL2	
SJAIDEN	DS	F	job identification on
*			non-BS2000 cluster
SJABUF@	DS	A	user buffer address
SJABUFL	DS	F	user buffer length
SJAOPT	DS	FL1	option
* _option_set			
SJAOTSN	EQU	0	*TSN
SJAOSVT	EQU	1	*SERVER-TSN
SJAOMJV	EQU	2	*MONJU
SJAOFRG	EQU	3	*FOREIGN
SJAOSYS	EQU	4	*SYSFILE
*			
SJAINFO	DS	AL1	information
SJADOPA	EQU	X'80'	document part info
SJADOFO	EQU	X'40'	document format info
SJAPRJC	EQU	X'20'	print job control info
SJALACO	EQU	X'10'	layout control info
SJAREDE	EQU	X'08'	resource description info
SJATOPR	EQU	X'04'	to printer info
SJAINFF	EQU	X'03'	
SJAUN3	DS	XL14	
SJA#	EQU	*-SJAHDR	

**Ausgabedatenbeschreibung für die Programmiersprache Assembler**

* _main_header_model				
SJA_MAIN_HEADER_MODEL	DS	OF		
SJAHULE	DS	F		length used in the block
SJAHDRB	DS	F		displacement to header block
SJADOPB	DS	F		displacement to document part
*				block
SJADOFF	DS	F		displacement to document
*				format block
SJAPJCB	DS	F		displacement to print job
*				control block
SJALCOB	DS	F		displacement to layout
*				control block
SJAREDB	DS	F		displacement to resource
*				description block
SJATOPB	DS	F		displacement to to printer
*				block
SJARSOB	DS	F		displacement to RSO main
*				block
SJASPSB	DS	F		displacement to SPS main
*				block
SJADSEB	DS	F		displacement to DSEM main
*				block
SJADPRB	DS	F		displacement to DPRINT main
*				block
SJAOUTPUT#	EQU		*-SJAHULE	



## SHOPJST - Status eines Druckauftrags anzeigen

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** C, D, E, L, M

Mit diesem Makro wird der Status eines Druckauftrags ausgegeben. Das Makro entspricht dem Kommando SHOW-PRINT-JOB-STATUS.

### Format

Operation	Operanden
SHOPJST	XPAND=PARAM / OUTPUT  ,VARIANT= <u>001</u> / <c-string 3..3>  ,CALLER= <u>*USER</u> / *SYSTEM  ,RSOPAR= <u>*NONE</u> / <var: pointer>  ,CLUSTER= <u>*LOCAL_CLUSTER</u> /<var: char: 8> / <c-string 1..8: name_1..8>  INFO = <var: enum-of _flag_set:1> / *ALL / *ORIGIN / *DESTINATION / *TRAITS / *DISTRIBUTED / *SUMMARY / *SPOOL-FILTER / *RSO-FILTER / default: *ORIGIN  ,BUFFER=<var: pointer>  ,BUFLEN=<var: int: 4> / <integer 1..8388607>  ,S_SPN_T= <u>*ALL</u> / <var: enum-of _flag_set:1> / *NONE / *WILD / *LIST  ,S_SPNAM= <u>*ALL</u> / array(16): <var: char: 8> / <c-string 1..8>  ,S_SPN_W= <u>*ALL</u> / <var: char: 24> / <c-string 1..24>

(Teil 1 von 6)

Operation	Operanden
SHOPJST	<p data-bbox="331 203 1227 265">,S_USR_T=<u>STD</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD / *OWN / *ALL</p> <p data-bbox="331 310 1227 341">,S_USER=<u>ALL</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p data-bbox="331 386 1227 417">,S_USERW=<u>ALL</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p data-bbox="331 462 1227 493">,S_HST_T=<u>HOME</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD / ALL</p> <p data-bbox="331 538 1227 569">,S_HOST=<u>ALL</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p data-bbox="331 614 1227 645">,S_HOSTW=<u>ALL</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p data-bbox="331 690 1227 753">,S_SRV_T=<u>STD</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD / *STD / *ALL / *HOME</p> <p data-bbox="331 797 1227 829">,S_SERV=<u>ALL</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p data-bbox="331 873 1227 905">,S_SERVW=<u>ALL</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p data-bbox="331 949 1227 981">,S_FRM_T=<u>ALL</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD</p> <p data-bbox="331 1025 1227 1057">,S_FORM=<u>ALL</u> / array(16): &lt;var: char: 6&gt; / &lt;c-string 1..6: c-string 1-6&gt;</p> <p data-bbox="331 1101 1227 1133">,S_FORMW=<u>ALL</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p data-bbox="331 1177 1227 1209">,S_TSN_T=<u>ALL</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST</p> <p data-bbox="331 1253 1227 1285">,S_TSN=<u>ALL</u> / array(16): &lt;var: char: 4&gt; / &lt;c-string 1..4: c-string 1-4&gt;</p> <p data-bbox="331 1329 1227 1361">,S_IDENT=<u>ALL</u> / &lt;var: int: 4&gt; / &lt;integer 1..2147483647&gt;</p> <p data-bbox="331 1405 1227 1437">,S_SRT_T=<u>ALL</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST</p> <p data-bbox="331 1481 1227 1513">,S_SVTSN=<u>ALL</u> / array(16): &lt;var: char: 4&gt; / &lt;c-string 1..4: c-string 1-4&gt;</p>

(Teil 2 von 6)

Operation	Operanden
SHOPJST	<p>,S_OVL_T=<u>*ALL</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST</p> <p>,S_OVER=<u>*ALL</u> / array(16): &lt;var: char: 2&gt; / &lt;c-string 2..2: c-string 2-2&gt;</p> <p>,S_SPC_T=<u>*ALL</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST</p> <p>,S_CLASS=<u>*ALL</u> / array(16): &lt;var: int: 1&gt; / &lt;integer 1..255&gt;</p> <p>,S_ACC_T=<u>*ALL</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST / *NONE</p> <p>,S_ACC=<u>*ALL</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>,S_FRT_T=<u>*ALL</u> / &lt;var:enum-of flag_set:1&gt;/*LIST</p> <p>,S_FRT_N=<u>*ALL</u> / array(16): &lt;var: char: 63&gt; / &lt;c-string 1..63: c-string 1-63&gt;</p> <p>,S_DESTY=<u>*ALL</u> / list-poss(8): *ALL / *DEVICE / *REMOTE / *MANAGED_DEV / *CENTRAL / *LOCAL / *DESTINATION</p> <p>,S_SELTY=<u>*MAY</u> / *MUST</p> <p>,S_SPDVT=<u>*ALL</u> / array(16): *ALL / *HP / *HP90 / *LP / *LP_EMUL / *LP48 / *LP65 / *TAPE / *2050_APA / *2090_APA / *2090_TWIN / *VIRTUAL / *PCL</p> <p>,S_DEV_T=<u>*OWN</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST / *WILD / *ALL</p> <p>,S_DVNAM=<u>*ALL</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>,S_DVNMW=<u>*ALL</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,S_DES_T=<u>*ALL</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD</p> <p>,S_DSNAM=<u>*ALL</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p>

(Teil 3 von 6)

Operation	Operanden
SHOPJST	<p>,S_DSNMW=<u>*ALL</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,S_ERR_T=<u>*ANY</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD / *ALL / *NONE</p> <p>,S_ERMSG =<u>*ANY</u> / array(16): &lt;var: char 1..8 / &lt;c-string 1..8&gt; / *ALL / *NONE</p> <p>,S_ERRMW =<u>*ANY</u> / &lt;var: char 1..24 / &lt;c-string 1..24&gt; / *ALL / *NONE</p> <p>,S_FOB=(<i>type, low, high</i>):  <i>type</i>: <u>*ANY</u> / *ONLY / *NO / *RANGE  <i>low</i>: <u>*MIN</u> / &lt;var: int: 2&gt; / &lt;integer 0..32767&gt;  <i>high</i>: <u>*MAX</u> / &lt;var: int: 2&gt; / &lt;integer 0..32767&gt;</p> <p>,S_CSN=(<i>type, low, high</i>):  <i>type</i>: <u>*ALL</u> / *ONE / *RANGE  <i>low</i>: <u>*MIN</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt;  <i>high</i>: <u>*MAX</u> / &lt;var: int: 2&gt; / &lt;integer 1..32767&gt;</p> <p>,S_PRIO=(<i>type, low, high</i>):  <i>type</i>: <u>*ALL</u> / *RANGE  <i>low</i>: <u>*MIN</u> / &lt;var: int: 1&gt; / &lt;integer 30..255&gt;  <i>high</i>: <u>*MAX</u> / &lt;var: int: 1&gt; / &lt;integer 30..255&gt;</p> <p>,S_ROT=<u>*ANY</u> / list-poss(4): *ANY / *YES / *NO</p> <p>,S_JTYP=<u>*ALL</u> / array(16): *ALL / *WAIT / *ACTIVE / *KEEP / *REPLAY / *WAIT_PRE / *PREPROC / *BEF_APA_PR / *AFT_APA_PR / *WAIT_FT / *FT</p> <p>,S_VPR_T=<u>*ALL</u> / *LIST / &lt;var: enum-of _flag_set:1&gt;</p> <p>,S_VPR_N=<u>*ALL</u> / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>,E_SPN_T=<u>*NONE</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD</p> <p>,E_SPNAM=<u>*NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p>

(Teil 4 von 6)

Operation	Operanden
SHOPJST	<p>,E_SPNAW=<u>NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,E_USR_T=<u>NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST / *WILD</p> <p>,E_USER=<u>NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8 c-string 1-8&gt;</p> <p>,E_USERW=<u>NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,E_HST_T=<u>NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST / *WILD</p> <p>,E_HOST=<u>NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>,E_HOSTW=<u>NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,E_SRV_T=<u>NONE</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD</p> <p>,E_SERV=<u>NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>,E_SERVW=<u>NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,E_FRM_T=<u>NONE</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD</p> <p>,E_FORM=<u>NONE</u> / array(16): &lt;var: char: 6&gt; &lt;c-string 1..6: c-string 1-6&gt;</p> <p>,E_FORMW=<u>NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p>,E_TSN_T=<u>NONE</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST</p> <p>,E_TSN=<u>NONE</u> / array(16): &lt;var: char: 4&gt; / &lt;c-string 1..4: c-string 1-4&gt;</p> <p>,E_SRT_T=<u>NONE</u> / &lt;var: enum-of _flag_set:1&gt; / *LIST</p> <p>,E_SVTSN=<u>NONE</u> / array(16): &lt;var: char: 4&gt; / &lt;c-string 1..4: c-string 1-4&gt;</p> <p>,E_OVL_T=<u>NONE</u> / &lt;var: enum-of _flag_set: 1&gt; *LIST</p>

(Teil 5 von 6)

Operation	Operanden
SHOPJST	<p data-bbox="328 204 1225 232">,E_OVER=<u>*NONE</u> / array(16): &lt;var: char: 2&gt; / &lt;c-string 2..2: c-string 2-2&gt;</p> <p data-bbox="328 277 1225 305">,E_SPC_T=<u>*NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST</p> <p data-bbox="328 350 1225 378">,E_CLASS=<u>*NONE</u> / array(16): &lt;var: int: 1&gt; / &lt;integer 1..255&gt;</p> <p data-bbox="328 422 1225 450">,E_ACC_T=<u>*NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST</p> <p data-bbox="328 495 1225 523">,E_ACC=<u>*NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p data-bbox="328 568 1225 596">,E_DESTY=<u>*NONE</u> / list-poss(4): *NONE / *DEVICE / *CENTRAL</p> <p data-bbox="328 640 1225 669">,E_DEV_T=<u>*NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST / *WILD</p> <p data-bbox="328 713 1225 741">,E_DVNAM=<u>*NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p data-bbox="328 786 1225 814">,E_DVNAW=<u>*NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p data-bbox="328 859 1225 887">,E_FRT_T=<u>*NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST</p> <p data-bbox="328 931 1225 992">,E_FRT_N=<u>*NONE</u> / array(16): &lt;var: char: 63&gt; / &lt;c-string 1..63: c-string 1-63&gt;</p> <p data-bbox="328 1004 1225 1032">,E_DES_T=<u>*NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST / *WILD</p> <p data-bbox="328 1077 1225 1105">,E_DSNAM=<u>*NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p data-bbox="328 1149 1225 1177">,E_DSNMW=<u>*NONE</u> / &lt;var: char: 24&gt; / &lt;c-string 1..24: c-string 1-24&gt;</p> <p data-bbox="328 1222 1225 1250">,E_VPR_T=<u>*NONE</u> / &lt;var: enum-of _flag_set: 1&gt; / *LIST</p> <p data-bbox="328 1295 1225 1323">,E_VPR_N=<u>*NONE</u> / array(16): &lt;var: char: 8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p>

(Teil 6 von 6)

## Operandenbeschreibung

### **XPAND=PARAM / OUTPUT**

Es wird festgelegt, welche Struktur zu expandieren (erzeugen) ist.

### **XPAND=PARAM**

Das Layout der Eingabe-Parameterliste wird expandiert

### **XPAND=OUTPUT**

Es werden alle Datenbeschreibungen (DSECTs) zur Beschreibung der Ausgabe-Informationenblöcke generiert.

### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **CALLER=\*USER / \*SYSTEM**

Aufrufer des Makros.

### **CALLER=\*USER**

Vorgabewert für diesen Operanden. Ein Benutzer ruft das Makro auf.

### **CALLER=\*SYSTEM**

Das Makro wird vom System aufgerufen

### **RSOPAR=\*NONE / <var: pointer>**

Es kann die Adresse einer Parameterliste angegeben werden, in der RSO-spezifische Werte für den Aufruf des Makros hinterlegt sind.

### **RSOPAR=\*NONE**

Die Werte sollen nicht über eine Parameterliste versorgt werden.

### **RSOPAR=<var: pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die RSO-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die RSO-Parameterliste hinterlegt ist (A(feld)).

Das Makro wird vom System aufgerufen.

### **CLUSTER=\*LOCAL CLUSTER / <var: char: 8> / <c-string 1..8: name\_1..8>**

Name des Clusters, in dem die Druckaufträge zu finden sind.

### **CLUSTER=\*LOCAL CLUSTER**

Vorgabewert für diesen Operanden. Die Druckaufträge werden im lokalen Cluster gesucht.

### **CLUSTER=<var: char: 8>**

Der Name des Clusters wird in einem Feld übergeben.

### **CLUSTER=<c-string 1..8: name\_1..8>**

Der Name des Clusters wird direkt angegeben.

**INFO=\*ORIGIN / <var: enum-of \_info\_set: 1> / \*DESTINATION / \*TRAITS / \*SUMMARY / \*DISTRIBUTED / \*SPOOL-FILTER / \*RSO-FILTER / \*ALL**

Mit diesem Operanden kann ausgewählt werden, welche Informationen ausgegeben werden sollen.

### **INFO=\*ORIGIN**

Gibt folgende Informationen über den Ursprung der Druckaufträge aus:

- TSN
- Name des Servers, auf dem der Auftrag verarbeitet wird
- Benutzerkennung
- Abrechnungsnummer des Benutzers, der den Druckauftrag erzeugt hat
- Anzahl der angegebenen Seitenkopien
- Anzahl der angegebenen Dateikopien
- SPOOL-NAME des Jobs
- Typ und Größe der Datei
- Name des Client-Hosts
- TSN des auftraggebenden Auftrags

Die ausgegebenen Informationen sind vor allem interessant für nichtprivilegierte Benutzer.

### **INFO=<var: enum-of \_info\_set: 1>**

Die Vereinbarung der nachgefragten Informationen wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und der gewünschten Information:

1	*ORIGIN
2	*DESTINATION
3	*TRAITS
4	*DEVICE_TYPE
5	*ALL
6	*SUMMARY
7	*DISTRIBUTED
8	*SPOOL-FILTER
9	*RSO-FILTER
siehe Beschreibung der Parameterliste ab <a href="#">Seite 347</a> .	



**INFO=\*DESTINATION**

Gibt Informationen über das Ausgabemedium der Druckaufträge aus:

- TSN
- Name des Servers, auf dem der Auftrag verarbeitet wird
- Art des Jobs (RSO oder lokaler SPOOL)
- Name des Geräts
- Name des Geräte-Pools
- Gerätetypen
- Status und Typ des Jobs (WAIT, ACTIVE, KEEP, ...)
- Errorcode und Fehlermeldung (RSO, Dprint)

Die ausgegebenen Informationen sind vor allem interessant für den Geräteverwalter.

**INFO=\*TRAITS**

Gibt Informationen über folgende Eigenschaften der Druckaufträge aus:

- Priorität
- Formularname
- SPOOLOUT-Klasse
- Anzahl der benötigten Zeichensätze
- Größe des benötigten FOB-Datendias
- Name des benötigten Filmdias
- Inanspruchnahme des Seitendrehmoduls
- Ausgabesteuerung (CONTROL)
- Name des Servers, auf dem der Auftrag verarbeitet wird
- Lage der Ressourcen

Die ausgegebenen Informationen sind vor allem interessant für die Systembetreuung, da sie sich auf die Eigenschaften der lokalen SPOOL-Geräte beziehen.

**INFO=\*SUMMARY**

Gibt die Anzahl der ausgewählten Druckaufträge und die Anzahl der für diese bereitgestellten PAM-Seiten aus.

**INFO=\*DISTRIBUTED**

Es werden globale Informationen über Druckaufträge ausgegeben:

- TSN am lokalen und fernen Host (Client- und Server-TSN)
- Name des Client-Hosts
- Name des Servers, auf dem der Auftrag verarbeitet wird
- Art des Jobs (lokaler, verteilt, zwischen Clustern)
- Informationen über die anfordernde Benutzerkennung

**INFO = \*SPOOL-FILTER / \*RSO-FILTER**

Diese Werte erlauben die Anzeige von Informationen über die Zielgeräte eines Druckauftrags. Die Liste der Gerätetypen entspricht den Gerätetypen, bei denen der Druckauftrag nach dem Filtern verarbeitet werden kann. Abhängig von den Filtern können die Druckaufträge von SPOOL (bei Angabe von INFO=\*SPOOL-FILTER) oder von RSO (bei Angabe von INFO=\*RSO-FILTER) verarbeitet werden.

Beide Werte können in Verbindung mit INFO=DESTINATION angegeben werden, wenn das Feld ERMMSG (Fehlermeldung) 'FILTER' oder 'DEL/FLT' enthält.

Ausgabe-Information:

- TSN
- Name des Servers, auf dem der Druckauftrag verarbeitet wird
- Auftragstyp (RSO oder lokaler SPOOL)
- Gerätename
- Druckerpoolname
- Gerätetyp
- Status und Typ des Auftrags (WAIT,ACTIVE,KEEP,...)
- Fehlercode
- Fehlermeldung

*Hinweis*

Diese Informationen sind in erster Linie für den Geräteverwalter und die Systembetreuung (PRINT SERVICE ADMINISTRATION) von Interesse.

**BUFFER=<var: pointer>**

Zeiger auf den Ausgabepuffer.

**BUFLEN=<var: int: 4> / <integer 1..8388607>**

Größe des Ausgabepuffers.

**BUFLEN=<var: int: 4>**

Die Größe des Ausgabepuffers wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**BUFLEN=<integer 1..8388607>**

Die Größe des Ausgabepuffers wird direkt angegeben.

**S\_SPN\_T=\*ALL / <var: enum-of \_flag\_set:1> / \*NONE / \*LIST / \*WILD**

Legt den Typ der Angabe für die SPOOLOUT-Namen der Druckaufträge fest.

**S\_SPN\_T=<var: enum-of \_flag\_set:1>**

Die Vereinbarung über den Typ der Angabe für S\_SPN\_T und S\_SPN\_W wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und dem gewünschten Typ:

0	*ALL/*ANY
1	*NONE
2	*LOCAL
3	*REMOTE
4	*CENTRAL
64	*DEVICE
128	*MANAG_DEV
5	*WILDCARD
6	*LIST
7	*STD
8	*OWN
9	*HOME
10	*IDENT
11	*ALL/*ANY
12	*ONLY
13	*RANGE
14	*ONE
15	*NO
16	*DESTINATION
siehe Beschreibung der Parameterliste ab <a href="#">Seite 347</a> .	

**S\_SPN\_T=\*NONE**

Es werden nur Druckaufträge ohne SPOOLOUT-Namen ausgewählt.

**S\_SPN\_T=\*ALL**

Alle SPOOLOUT-Namen der Druckaufträge werden ausgewählt.

**S\_SPN\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_SPNAM eine Liste von SPOOLOUT-Namen enthält.

**S\_SPN\_T=\*WILD**

Mit \*WILD legen Sie fest, dass der Operand S\_SPN\_W einen mit Platzhaltern teilqualifizierten Namen enthält. Werte in S\_SPNAM erzeugen in diesem Fall einen Fehler.

**S\_SPNAM=\*ALL / array(16): <var: char: 8> / <c-string 1.8>**

SPOOLOUT-Namen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_SPNAM=\*ALL**

Alle SPOOLOUT-Namen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_SPNAM=array(16): <var: char: 8>**

Liste der SPOOLOUT-Namen der Druckaufträge, über die Informationen ausgegeben werden sollen. S\_SPN\_T=\*LIST muss dafür gesetzt sein.

**S\_SPNAM=<c-string 1..8>**

SPOOLOUT-Name des Druckauftrags, über den Informationen ausgegeben werden sollen.

**S\_SPN\_W=\*ALL / <var: char: 24> / <c-string 1..24>**

SPOOLOUT-Namen der Druckaufträge, über die Informationen ausgegeben werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben. S\_SPN\_T=\*WILD muss dafür gesetzt sein.

**S\_SPN\_W=\*ALL**

Alle SPOOLOUT-Namen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_SPN\_W=<var: char: 24> / <c-string 1..24>**

SPOOLOUT-Namen der Druckaufträge, über die Informationen ausgegeben werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_USR\_T=\*STD / <var: enum-of \_flag\_set:1> / \*LIST / \*WILD / \*OWN / \*ALL**

Legt den Typ der Angabe für die Benutzerkennungen der Druckaufträge fest.

**S\_USR\_T=\*STD**

\*STD bedeutet \*ALL für Operator-Tasks und \*OWN für alle anderen Tasks.

**S\_USR\_T=<var: enum-of \_flag\_set:1>**

Die Vereinbarung über den Typ der Angabe für S\_USER und S\_USERW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_USR\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_USER eine Liste von Benutzerkennungen enthält.

**S\_USR\_T=\*WILD**

Mit \*WILD legen Sie fest, dass der Operand S\_USERW einen mit Platzhaltern teilqualifizierten Namen enthält. Werte in S\_USER erzeugen in diesem Fall einen Fehler.

**S\_USR\_T=\*OWN**

Nur eigene Druckaufträge werden ausgewählt.

**S\_USR\_T=\*ALL**

Alle Druckaufträge werden ausgewählt.

**S\_USER=\*ALL / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_USER=\*ALL**

Alle Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_USER=array(16): <var: char: 8>**

Liste der Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_USER=<c-string 1..8: c-string 1-8>**

Benutzerkennung des Druckauftrags, über den Informationen ausgegeben werden sollen.

**S\_USERW=\*ALL / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_USERW=\*ALL**

Alle Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_USERW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Mit Platzhaltern teilqualifizierte Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_HST\_T=\*HOME / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD / \*ALL**

Legt den Typ der Angabe für die Namen der Host-Rechner fest.

**S\_HST\_T=\*HOME**

Mit \*HOME legen Sie fest, dass die Abfrage für den lokalen Rechner gilt.

**S\_HST\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_HOST und S\_HOSTW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_HST\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_HOST eine Liste von Rechnernamen enthält.

**S\_HST\_T=\*WILD**

Mit \*WILD legen Sie fest, dass der Operand S\_HOSTW einen mit Platzhaltern teilqualifizierten Namen enthält. Werte in S\_HOST erzeugen in diesem Fall einen Fehler.

**S\_HST\_T=\*ALL**

Alle Namen der Host-Rechner werden ausgewählt.

**S\_HOST=\*ALL / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Es werden Informationen über Druckaufträge ausgegeben, die von den angegebenen Hosts gestartet wurden. Nichtprivilegierte Benutzer erhalten bei der Spezifikation eines fernen Hosts keine Informationen, da sie nur über ihre eigenen Druckaufträge informiert werden.

**S\_HOST=\*ALL**

Alle Namen der Host-Rechner werden ausgewählt.

**S\_HOST=array(16): <var: char: 8>**

Es werden Informationen über Druckaufträge ausgegeben, die von den in der Liste angegebenen Hosts gestartet wurden.

**S\_HOST=<c-string 1..8: c-string 1-8>**

Es werden Informationen über Druckaufträge ausgegeben, die von dem angegebenen Host gestartet wurden.

**S\_HOSTW=\*ALL / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Es werden Informationen über Druckaufträge ausgegeben, die von den angegebenen Hosts gestartet wurden. Nichtprivilegierte Benutzer erhalten bei der Spezifikation eines fernen Hosts keine Informationen, da sie nur über ihre eigenen Druckaufträge informiert werden.

Die Namen der Rechner werden mit Platzhaltern teilqualifiziert angegeben.

**S\_HOSTW=\*ALL**

Alle Namen der Host-Rechner werden ausgewählt.

**S\_HOSTW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Namen der Rechner werden mit Platzhaltern teilqualifiziert angegeben.

**S\_SRV\_T=\*HOME / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD / \*ALL**

Legt den Typ der Angabe für die Namen der Server fest.

**S\_SRV\_TYP=\*STD/<var: enum-of \_flag\_set:1>/\*LIST/\*WILD/\*ALL/\*HOME**

Legt den Typ des Eintrags für den Server-Namen fest.

**S\_SRV\_TYP=\*STD**

\*STD bedeutet \*ALL für Inter-Cluster- und \*HOME für Intra-Cluster-Anfragen.

**S\_SRV\_T=\*HOME**

Mit \*HOME legen Sie fest, dass die Abfrage für den lokalen Server gilt.

**S\_SRV\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_SERV und S\_SERVW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_SRV\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_SRV eine Liste von Rechnernamen enthält.

**S\_SRV\_T=\*WILD**

Mit \*WILD legen Sie fest, dass der Operand S\_SRVW einen mit Platzhaltern teilqualifizierten Namen enthält. Werte in S\_SRV erzeugen in diesem Fall einen Fehler.

**S\_SRV\_T=\*ALL**

Alle Namen der Server-Rechner werden ausgewählt.

**S\_SRV=\*ALL / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Es sollen Informationen über Druckaufträge ausgegeben werden, die auf dem angegebenen Server verarbeitet werden.

**S\_SRV=\*ALL**

Alle Namen der Server-Rechner werden ausgewählt.

**S\_SRV=array(16): <var: char: 8>**

Es sollen Informationen über Druckaufträge ausgegeben werden, die auf den in der Liste angegebenen Servern verarbeitet werden.

**S\_SRV=<c-string 1..8: c-string 1-8>**

Es sollen Informationen über Druckaufträge ausgegeben werden, die auf dem angegebenen Server verarbeitet werden.

**S\_SRVW=\*ALL / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Es sollen Informationen über Druckaufträge ausgegeben werden, die auf dem angegebenen Server verarbeitet werden. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_FRM\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für die Namen der Formularnamen fest.

Mit \*LIST legen Sie fest, dass der Operand S\_FORM eine Liste von Formularnamen enthält.

Mit \*WILD legen Sie fest, dass der Operand S\_FORMW einen mit Platzhaltern teilqualifizierten Namen enthält. Werte in S\_FORM erzeugen in diesem Fall einen Fehler.

**S\_FRM\_T=\*ALL**

Alle Formularnamen werden ausgewählt.

**S\_FRM\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_FORM und S\_FORMW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_FRM\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_FORM eine Liste von Formularnamen enthält.

**S\_FRM\_T=\*WILD**

Mit \*WILD legen Sie fest, dass der Operand S\_FORMW einen mit Platzhaltern teilqualifizierten Namen enthält. Werte in S\_FORM erzeugen in diesem Fall einen Fehler.

**S\_FORM=\*ALL / array(16): <var: char: 6> / <c-string 1..6: c-string 1-6>**

Formularnamen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_FORM=\*ALL**

Alle Formularnamen werden ausgewählt.

**S\_FORM=array(16): <var: char: 6>**

Liste der Formularnamen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_FORM=<c-string 1..6: c-string 1-6>**

Formularname der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_FORMW=\*ALL / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Formularnamen der Druckaufträge, über die Informationen ausgegeben werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_FORMW=\*ALL**

Alle Formularnamen werden ausgewählt.

**S\_FORMW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Mit Platzhaltern teilqualifizierte Formularnamen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_TSN\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Auftragsnummern der Druckaufträge fest.

**S\_TSN\_T=\*ALL**

Alle Auftragsnummern werden ausgewählt.

**S\_TSN\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_TSN wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_TSN\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_TSN eine Liste von Auftragsnummern enthält.



**S\_TSN=\*ALL / array(16): <var: char: 4> / <c-string 1..4: c-string 1-4>**

Auftragsnummern der Druckaufträge, über die informiert werden soll.

**S\_TSN=\*ALL**

Alle Auftragsnummern werden ausgewählt.

**S\_TSN=array(16): <var: char: 4>**

Liste der Auftragsnummern der Druckaufträge, über die informiert werden soll.

**S\_TSN=<c-string 1..4: c-string 1-4>**

Auftragsnummer des Druckauftrags, über die informiert werden soll.

**S\_IDENT=\*ALL / <var: int: 4> / <integer 1..2147483647>**

Es können Druckaufträge in UNIX-Systemen über ihre Identifikation angesprochen werden. In diesem Fall muss eine Xprint-Domäne im UNIX-System ausgewählt werden. Näheres siehe Handbuch „[Distributed Print Services \(BS2000/OSD\)](#)“.

**S\_IDENT=\*ALL**

Alle Druckaufträge in UNIX-Systemen werden angesprochen.

**S\_IDENT=<var: int: 4> / <integer 1..2147483647>**

Es werden Druckaufträge in UNIX-Systemen über ihre Identifikation angesprochen. In diesem Fall muss eine Xprint-Domäne im UNIX-System ausgewählt werden.

**S\_SRT\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Server-TSN der Druckaufträge fest.

**S\_SRT\_T=\*ALL**

Alle Server-TSN werden ausgewählt.

**S\_SRT\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_SVTSN wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_SRT\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_SVTSN eine Liste von Server-TSN enthält.

**S\_SVTSN=\*ALL / array(16): <var: char: 4> / <c-string 1..4: c-string 1-4>**

Server-TSN der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_SVTSN=\*ALL**

Alle Server-TSN werden ausgewählt.

**S\_SVTSN=array(16): <var: char: 4>**

Liste der Server-TSN der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_SVTSN=<c-string 1..4: c-string 1-4>**

Server-TSN der Druckaufträge, über die Informationen ausgegeben werden sollen.

**S\_OVL\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Filmdias der Druckaufträge fest.

**S\_OVL\_T=\*ALL**

Alle Filmdias werden ausgewählt.

**S\_OVL\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_OVER wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_OVL\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_OVER eine Liste von Filmdias enthält.

**S\_OVER=\*ALL / array(16): <var: char: 2> / <c-string 2..2: c-string 2-2>**

Informationen über Druckaufträge, für die die genannten Filmdias angegeben wurden, sollen ausgegeben werden.

**S\_OVER=\*ALL**

Alle Filmdias werden ausgewählt.

**S\_OVER=array(16): <var: char: 2>**

Informationen über Druckaufträge, für die die in der Liste genannten Filmdias angegeben wurden, sollen ausgegeben werden.

**S\_OVER=<c-string 2..2: c-string 2-2>**

Informationen über Druckaufträge, für die das genannte Filmdia angegeben wurde, sollen ausgegeben werden.

**S\_SPC\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die SPOOLOUT-Klassen der Druckaufträge fest.

**S\_SPC\_T=\*ALL**

Alle SPOOLOUT-Klassen werden ausgewählt.

**S\_SPC\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_CLASS wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_SPC\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_CLASS eine Liste von SPOOLOUT-Klassen enthält.

**S\_CLASS=\*ALL / array(16): <var: int: 1> / <integer 1..255>**

SPOOLOUT-Klassen der Druckaufträge, über die informiert werden soll.

**S\_CLASS=\*ALL**

Alle SPOOLOUT-Klassen werden ausgewählt.

**S\_CLASS=array(16): <var: int: 1>**

Liste der SPOOLOUT-Klassen der Druckaufträge, über die informiert werden soll.

**S\_CLASS=<integer 1..255>**

Die SPOOLOUT-Klasse der Druckaufträge, über die informiert werden soll.

**S\_ACC\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST / \*NONE**

Legt den Typ der Angabe für die Abrechnungsnummern der Druckaufträge fest.

**S\_ACC\_T=\*ALL**

Alle Abrechnungsnummern werden ausgewählt.

**S\_ACC\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_ACC wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_ACC\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_ACC eine Liste von Abrechnungsnummern enthält.

**S\_ACC\_T=\*NONE**

Keine Abrechnungsnummer wird ausgewählt.

**S\_ACC=\*ALL / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Abrechnungsnummern der Druckaufträge, über die informiert werden soll.

**S\_ACC=\*ALL**

Alle Abrechnungsnummern werden ausgewählt.

**S\_ACC=array(16): <var: char: 8>**

Liste der Abrechnungsnummern der Druckaufträge, über die informiert werden soll.

**S\_ACC=<c-string 1..8: c-string 1-8>**

Abrechnungsnummer der Druckaufträge, über die informiert werden soll.

**S\_FRT\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Formatnamen der Druckaufträge fest.

**S\_FRT\_T=\*ALL**

Alle Formatnamen werden ausgewählt.

**S\_FRT\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_FRT wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_FRT\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand S\_FRT\_N eine Liste von Formatnamen enthält.

**S\_FRT\_N=\*ALL / array(16): <var: char : 63> / <c-string 1..63: c-string 1-63>**

Formatnamen der Druckaufträge, über die informiert werden soll.

**S\_FRT\_N=\*ALL**

Alle Formatnamen werden ausgewählt.

**S\_FRT\_N=array(16): <var: char : 63>**

Liste der Formatnamen der Druckaufträge, über die informiert werden soll.

**S\_FRT\_N=<c-string 1..63: c-string 1-63>**

Formatname des Druckauftrags, über den informiert werden soll.

**S\_DESTY=\*ALL / list-poss(8): \*ALL / \*DEVICE / \*REMOTE / \*MANAGED\_DEV / \*CENTRAL / \*LOCAL / \*DESTINATION**

Es sollen Informationen über die Druckaufträge ausgegeben werden, die auf den angegebenen Ausgabemedien verarbeitet werden. Es können Informationen angefordert werden über:

- alle Jobs (\*ALL),
- Jobs für lokale SPOOL-Geräte (\*LOCAL),
- Jobs für RSO-Geräte (\*REMOTE),
- Jobs ohne Zielangabe, d.h. deren Bestimmungsort der zentrale Druckerpool ist (\*CENTRAL),
- Jobs für Geräte, die vom Anwender verwaltet werden (\*MANAGED-DEVICES)
- oder bestimmte, hier angegebene Geräte (\*DEVICE).

**S\_SELTY=\*MAY / list-poss(3): \*MAY / \*MUST**

Die Auswahl der lokalen Druckaufträge wird eingegrenzt auf diejenigen, die zumindest auch auf einem der angegebenen Gerätetypen ausgegeben werden dürfen (\*MAY) oder auf diejenigen, die auf einem der angegebenen Gerätetypen ausgegeben werden müssen, d.h. auf keinem anderen als den angegebenen ausgegeben werden dürfen (\*MUST).

**S\_SPDVT=\*ALL / array(16): \*ALL / \*HP / \*HP90 / \*LP / \*LP\_EMUL / \*LP48 / \*LP65 / \*TAPE / \*2050\_APA / \*2090\_APA / \*2090\_TWIN / \*VIRTUAL**

Gerätetypen für lokale Druckaufträge, über die Informationen angefordert werden.

**S\_DEV\_T=\*OWN / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD / \*ALL**

Legt den Typ der Angabe für die Gerätenamen fest.

**S\_DEV\_T=\*OWN**

Nur eigene Gerätenamen werden angegeben.

**S\_DEV\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_DVNAME und S\_DVNMW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_DEV\_T=\*LIST**

Eine Liste von Gerätenamen wird angegeben.

**S\_DEV\_T=\*WILD**

Die Gerätenamen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_DEV\_T=\*ALL**

Alle Gerätenamen sind gültig.

**S\_DVNAME=\*ALL / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Namen der Geräte, über die Informationen angefordert werden.

**S\_DVNAME=\*ALL**

Über alle Geräte werden Informationen angefordert.

**S\_DVNAME=array(16): <var: char: 8>**

Namensliste der Geräte, über die Informationen angefordert werden.

**S\_DVNAME=<c-string 1..8: c-string 1-8>**

Name des Geräts, über das Informationen angefordert werden.

**S\_DVNMW=\*ALL / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Namen der Geräte, über die Informationen angefordert werden. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_DVNMW=\*ALL**

Über alle Geräte werden Informationen angefordert.

**S\_DVNMW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Namen der Geräte, über die Informationen angefordert werden. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_DES\_T=\*ALL / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für die Gerätenamen fest.

**S\_DES\_T=\*ALL**

Alle Gerätenamen sind gültig.

**S\_DES\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für S\_DSNAM und S\_DSNMW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**S\_DES\_T=\*LIST**

Eine Liste von Gerätenamen wird angegeben.

**S\_DES\_T=\*WILD**

Die Gerätenamen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_DSNAM=\*ALL / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Namen der Geräte-Pools, über die Informationen angefordert werden.

**S\_DSNAM=\*ALL**

Über alle Geräte-Pools werden Informationen angefordert.

**S\_DSNAM=array(16): <var: char: 8>**

Namensliste der Geräte-Pools, über die Informationen angefordert werden.

**S\_DSNAM=<c-string 1..8: c-string 1-8>**

Name des Geräte-Pools, über den Informationen angefordert werden.

**S\_DSNMW=\*ALL / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Namen der Geräte-Pools, über die Informationen angefordert werden. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_DSNMW=\*ALL**

Über alle Geräte-Pools werden Informationen angefordert.

**S\_DSNMW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Namen der Geräte-Pools, über die Informationen angefordert werden. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

**S\_ERR\_T=\*ANY / <var: enum-of FLAG\_SET:1> / \*LIST / \*WILD / \*ALL / \*NONE**

Legt den Typ des Eintrags für die Auswahl des Fehlermeldungsoperanden fest.

**S\_ERR\_T=\*ANY**

Alle Druckaufträge werden ausgewählt, egal ob sie eine entsprechende Fehlermeldung haben oder nicht.

**S\_ERR\_T=\*WILD**

Mit \*WILD wird festgelegt, dass der Operand S\_ERRMW mit Platzhaltern teilqualifiziert ist. Wird im Feld S\_ERMSG ein Wert angegeben, führt dies zu einem Fehler.

**S\_ERR\_T=\*LIST**

Mit \*LIST wird festgelegt, dass S\_ERMSG eine Liste von Fehlermeldungen enthält.

**S\_ERR\_T=\*ALL**

Alle gegenwärtigen Druckaufträge mit einer Fehlermeldung werden ausgewählt.

**S\_ERR\_T=\*NONE**

Alle gegenwärtigen Druckaufträge mit keiner Fehlermeldung werden ausgewählt.

**S\_ERMSG=\*ANY / \*ALL / \*NONE / array(16) <var: char 1..8> / <c-string 1..8>**

Wählt die Druckaufträge aus, die von den angegebenen Fehlermeldungen betroffen sind.

**S\_ERMSG=\*ANY**

Alle Druckaufträge werden ausgewählt, egal ob sie eine entsprechende Fehlermeldung haben oder nicht.

**S\_ERMSG=\*ALL**

Alle gegenwärtigen Druckaufträge mit einer Fehlermeldung werden ausgewählt.

**S\_ERMSG=\*NONE**

Alle gegenwärtigen Druckaufträge ohne Fehlermeldung werden ausgewählt.

**S\_ERMSG=array(16) <var: char 1..8> / <c-string 1..8>**

Die Druckaufträge, die von der angegebenen Liste von Fehlermeldungen betroffen sind, werden ausgewählt.

**S\_ERRMW=\*ANY/<var: char 1..24>/<c-string 1..24>/\*ALL/\*NONE**

Wählt die Druckaufträge aus, die von den angegebenen Fehlermeldungen betroffen sind. Die Fehlermeldungen können dabei mit Platzhaltern teilqualifiziert sein, allerdings muss dafür S\_ERR\_T=\*WILD eingestellt sein.

**S\_ERRMW=\*ANY**

Alle Druckaufträge werden ausgewählt, egal ob sie eine entsprechende Fehlermeldung haben oder nicht.

**S\_ERRMW=\*ALL**

Alle gegenwärtigen Druckaufträge mit einer Fehlermeldung werden ausgewählt.

**S\_ERRMW=\*NONE**

Alle gegenwärtigen Druckaufträge ohne Fehlermeldung werden ausgewählt.

**S\_ERRMW=<var: char 1..24>/<c-string 1..24>**

Fehlermeldungen, die den Druckauftrag betreffen, werden ausgewählt. Die Fehlermeldungen können dabei mit Platzhaltern teilqualifiziert sein, allerdings muss dafür S\_ERR\_T=\*WILD eingestellt sein.

**S\_FOB=(type, low, high):**

Informationen über Druckaufträge sollen ausgegeben werden,

- für die ein beliebiges FOB-Datendia angegeben wurde (\*ONLY)
- für die kein FOB-Datendia angegeben wurde (\*NO)
- für die FOB-Datendias im spezifizierten Größenbereich angegeben wurden (\*RANGE).

Bei Angabe von \*ANY wirkt der Operand nicht als Auswahlkriterium.

*type:* **\*ANY / \*ONLY / \*NO / \*RANGE**

Legt den Typ der Angabe für die Auswahl eines FOB-Datendia fest.

*low:* **\*MIN / <var: int: 2> / <integer 0..32767>**

Legt die Mindestgröße des FOB-Datendias fest. \*MIN entspricht dabei dem Wert 0.

*high:* **\*MAX / <var: int: 2> / <integer 0..32767>**

Legt die Maximalgröße des FOB-Datendias fest. \*MAX entspricht dabei dem Wert 32767.

**S\_CSN=(type, low, high):**

Informationen über Druckaufträge sollen ausgegeben werden,

- die nur einen Zeichensatz benötigen (\*ONE)
- deren Anzahl benötigter Zeichensätze sich im angegebenen Bereich befindet (\*RANGE).

*type:* **\*ALL / \*ONE / \*RANGE**

Legt den Typ der Angabe für die Auswahl der Zeichensätze fest.

*low:* **\*MIN / <var: int: 2> / <integer 1..32767>**

Legt die Mindestanzahl an Zeichensätzen fest. \*MIN entspricht dabei dem Wert 1.

*high:* **\*MAX / <var: int: 2> / <integer 1..32767>**

Legt die Maximalanzahl an Zeichensätzen fest. \*MAX entspricht dabei dem Wert 32767.

**S\_PRIO=(type, low, high):**

Informationen über Druckaufträge, deren Priorität sich im angegebenen Bereich befindet (\*RANGE), sollen ausgegeben werden. Bei Angabe von \*ALL wirkt der Operand nicht als Auswahlkriterium.

*type:* **\*ALL / \*RANGE**

Legt den Typ der Angabe für die Priorität fest.

*low:* **\*MIN / <var: int: 1> / <integer 30..255>**

Legt den niedrigsten Wert der Priorität fest.

*high:* **\*MAX / <var: int: 1> / <integer 30..255>**

Legt den höchsten Wert der Priorität fest.



**S\_ROT=\*ANY / list-poss(4): \*ANY / \*YES / \*NO**

Informationen über Druckaufträge,

- für die Seitendrehung im PRINT-DOCUMENT angefordert wurde (\*YES),
  - für die keine Seitendrehung im PRINT-DOCUMENT angefordert wurde (\*NO),
- sollen ausgegeben werden.

Bei Angabe von \*ANY wirkt der Operand nicht als Auswahlkriterium.

**S\_JTYP=\*ALL / array(16): \*ALL / \*WAIT / \*ACTIVE / \*KEEP / \*REPLAY / \*WAIT\_PRE / \*PREPROC / \*BEF\_APA\_PR / \*AFT\_APA\_PR / \*WAIT\_FT / \*FT**

Informationen über folgende Typen von Druckaufträge sollen ausgegeben werden:

- Wartende Druckaufträge: \*WAIT
- Aktive Druckaufträge: \*ACTIVE
- Angehaltene Druckaufträge: \*KEEP
- Auf Replayband auszugebende Druckaufträge: \*REPLAY
- Wartende SECTION-Druckaufträge (abschnittsweise Ausgaben):  
\*WAIT-PREPROCESSING
- Aktive SECTION-Druckaufträge (abschnittsweise Ausgaben): PREPROCESSING
- \*BEFORE-APA (APA-Drucker: Job-Status=„TRANSFERRED“, siehe auch Handbuch „[SPOOL \(BS2000/OSD\)](#)“).
- \*AFTER-APA (APA-Drucker: Job-Status=„TRANSIT“, siehe auch Handbuch „[SPOOL \(BS2000/OSD\)](#)“).
- Auf Filetransfer wartende Aufträge: \*WAIT-FILE-TRANSFER
- Aufträge, die aktuell mit Filetransfer übertragen werden: \*FILE-TRANSFER

**E\_SPN\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für die SPOOLOUT-Namen der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SPN\_T=\*NONE**

Kein SPOOLOUT-Name wird von der Informationsausgabe ausgeschlossen.

**E\_SPN\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_SPNAM und E\_SPNAW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_SPN\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_SPNAM eine Liste von SPOOLOUT-Namen enthält.

**E\_SPN\_T=\*WILD**

Die SPOOLOUT-Namen werden mit Platzhaltern teilqualifiziert angegeben.

**E\_SPNAM=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

SPOOLOUT-Namen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SPNAM=\*NONE**

Kein SPOOLOUT-Name wird von der Informationsausgabe ausgeschlossen.

**E\_SPNAM=array(16): <var: char: 8>**

Liste der SPOOLOUT-Namen, über die keine Informationen ausgegeben werden sollen.

E\_SPN\_T=\*LIST muss dafür gesetzt sein.

**E\_SPNAM=<c-string 1..8: c-string 1-8>**

SPOOLOUT-Name, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_SPNAW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

SPOOLOUT-Namen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SPNAW=\*NONE**

Kein SPOOLOUT-Name wird von der Informationsausgabe ausgeschlossen.

**E\_SPNAW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die SPOOLOUT-Namen werden mit Platzhaltern teilqualifiziert angegeben.

**E\_USR\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für die Benutzerkennungen der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_USR\_T=\*NONE**

Keine Benutzerkennung wird von der Informationsausgabe ausgeschlossen.

**E\_USR\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_USER und E\_USERW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_USR\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_USER eine Liste von Benutzerkennungen enthält.

**E\_USR\_T=\*WILD**

Die Benutzerkennungen werden mit Platzhaltern teilqualifiziert angegeben.

**E\_USER=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Benutzerkennungen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_USER=\*NONE**

Keine Benutzerkennung wird von der Informationsausgabe ausgeschlossen.

**E\_USER=array(16): <var: char: 8>**

Liste der Benutzerkennungen, über die keine Informationen ausgegeben werden sollen.

E\_USR\_T=\*LIST muss dafür gesetzt sein.

**E\_USER=<c-string 1..8: c-string 1-8>**

Benutzerkennung, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_USERW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Benutzerkennungen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen. Die Benutzerkennungen werden mit Platzhaltern teilqualifiziert angegeben.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_USERW=\*NONE**

Keine Benutzerkennung wird von der Informationsausgabe ausgeschlossen.

**E\_USERW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Benutzerkennungen werden mit Platzhaltern teilqualifiziert angegeben.

**E\_HST\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für Hosts fest, von denen Druckaufträge gestartet wurden, die von der Informationsausgabe ausgeschlossen werden sollen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_HST\_T=\*NONE**

Kein Host wird von der Informationsausgabe ausgeschlossen.

**E\_HST\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_HOST und E\_HOSTW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_HST\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_HOST eine Liste von Benutzerkennungen enthält.

**E\_HST\_T=\*WILD**

Die Hosts werden mit Platzhaltern teilqualifiziert angegeben.

**E\_HOST=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Druckaufträge, die von den angegebenen Hosts gestartet wurden, sind von der Informationsausgabe ausgeschlossen. Angegebene Hosts müssen zum lokalen Cluster gehören. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_HOST=\*NONE**

Kein Host wird von der Informationsausgabe ausgeschlossen.

**E\_HOST=array(16): <var: char: 8>**

Liste der Hosts, über die keine Informationen ausgegeben werden sollen. E\_HST\_T=\*LIST muss dafür gesetzt sein.

**E\_HOST=<c-string 1..8: c-string 1-8>**

Host, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_HOSTW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Druckaufträge, die von den angegebenen Hosts gestartet wurden, sind von der Informationsausgabe ausgeschlossen. Angegebene Hosts müssen zum lokalen Cluster gehören. Die Namen der Hosts werden mit Platzhaltern teilqualifiziert angegeben. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_HOSTW=\*NONE**

Kein Host wird von der Informationsausgabe ausgeschlossen.

**E\_HOSTW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Hosts werden mit Platzhaltern teilqualifiziert angegeben.

**E\_SRV\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für Server fest, von denen Druckaufträge verarbeitet werden sollen, die von der Informationsausgabe ausgeschlossen sind. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SRV\_T=\*NONE**

Kein Server wird von der Informationsausgabe ausgeschlossen.

**E\_SRV\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_SERV und E\_SERVW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_SRV\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_SERV eine Liste von Benutzerkennungen enthält.

**E\_SRV\_T=\*WILD**

Die Server werden mit Platzhaltern teilqualifiziert angegeben.

**E\_SERV=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Druckaufträge, die auf dem angegebenen Server verarbeitet werden sollen, sind von der Informationsausgabe ausgeschlossen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SERV=\*NONE**

Kein Server wird von der Informationsausgabe ausgeschlossen.

**E\_SERV=array(16): <var: char: 8>**

Liste der Server, über die keine Informationen ausgegeben werden sollen.

E\_SRV\_T=\*LIST muss dafür gesetzt sein.

**E\_SERV=<c-string 1..8: c-string 1-8>**

Server, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_SERVW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Druckaufträge, die auf den angegebenen Server verarbeitet werden sollen, sind von der Informationsausgabe ausgeschlossen. Die Namen der Server werden mit Platzhaltern teilqualifiziert angegeben.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SERVW=\*NONE**

Kein Server wird von der Informationsausgabe ausgeschlossen.

**E\_SERVW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Server werden mit Platzhaltern teilqualifiziert angegeben.

**E\_FRM\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für Formularnamen der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_FRM\_T=\*NONE**

Kein Formularnamen wird von der Informationsausgabe ausgeschlossen.

**E\_FRM\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_FORM und E\_FORMW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_FRM\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_FORM eine Liste von Benutzerkennungen enthält.

**E\_FRM\_T=\*WILD**

Die Formularnamen werden mit Platzhaltern teilqualifiziert angegeben.

**E\_FORM=\*NONE / array(16): <var: char: 6> / <c-string 1..6: c-string 1-6>**

Formularnamen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_FORM=\*NONE**

Kein Formularnamen wird von der Informationsausgabe ausgeschlossen.

**E\_FORM=array(16): <var: char: 6>**

Liste der Formularnamen, über die keine Informationen ausgegeben werden sollen.

E\_FRM\_T=\*LIST muss dafür gesetzt sein.

**E\_FORM=<c-string 1..6: c-string 1-6>**

Formularname, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_FORMW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Formularnamen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen. Die Formularnamen werden mit Platzhaltern teilqualifiziert angegeben.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_FORMW=\*NONE**

Kein Formularnamen wird von der Informationsausgabe ausgeschlossen.

**E\_FORMW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Formularnamen werden mit Platzhaltern teilqualifiziert angegeben.

**E\_TSN\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für Auftragsnummern der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_TSN\_T=\*NONE**

Keine Auftragsnummer wird von der Informationsausgabe ausgeschlossen.

**E\_TSN\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_TSN wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_TSN\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_TSN eine Liste von Benutzerkennungen enthält.

**E\_TSN=\*NONE / array(16): <var: char: 4> / <c-string 1..4: c-string 1-4>**

Auftragsnummern der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_TSN=\*NONE**

Keine Auftragsnummer wird von der Informationsausgabe ausgeschlossen.

**E\_TSN=array(16): <var: char: 4>**

Liste der Auftragsnummern, über die keine Informationen ausgegeben werden sollen.

E\_TSN\_T=\*LIST muss dafür gesetzt sein.

**E\_TSN=<c-string 1..4: c-string 1-4>**

Auftragsnummer, die von der Informationsausgabe ausgeschlossen werden soll.

**E\_SRT\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Server-TSN fest.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SRT\_T=\*NONE**

Keine Server-TSN wird von der Informationsausgabe ausgeschlossen.

**E\_SRT\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_SVTSN wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_SRT\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_SVTSN eine Liste von Benutzerkennungen enthält.

**E\_SVTSN=\*NONE / array(16): <var: char: 4> / <c-string 1..4: c-string 1-4>**

Druckaufträge mit der angegebenen Server-TSN sind von der Informationsausgabe ausgeschlossen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SVTSN=\*NONE**

Keine Server-TSN wird von der Informationsausgabe ausgeschlossen.

**E\_SVTSN=array(16): <var: char: 4>**

Liste der Server-TSN, über die keine Informationen ausgegeben werden sollen.

E\_SRT\_T=\*LIST muss dafür gesetzt sein.

**E\_SVTSN=<c-string 1..4: c-string 1-4>**

Server-TSN, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_OVL\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für Filmdias fest.

**E\_OVL\_T=\*NONE**

Kein Filmdia wird von der Informationsausgabe ausgeschlossen.

**E\_OVL\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_OVER wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_OVL\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_OVER eine Liste von Benutzerkennungen enthält.

**E\_OVER=\*NONE / array(16): <var: char: 2> / <c-string 2..2: c-string 2-2>**

Druckaufträge, für die diese Filmdias angegeben wurden, sollen von der Informationsausgabe ausgeschlossen werden.

**E\_OVER=\*NONE**

Kein Filmdia wird von der Informationsausgabe ausgeschlossen.

**E\_OVER=array(16): <var: char: 2>**

Liste der Filmdias, über die keine Informationen ausgegeben werden sollen.

E\_OVL\_T=\*LIST muss dafür gesetzt sein.

**E\_OVER=<c-string 2..2: c-string 2-2>**

Filmdia, das von der Informationsausgabe ausgeschlossen werden soll.

**E\_SPC\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für SPOOLOUT-Klassen der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_SPC\_T=\*NONE**

Keine SPOOLOUT-Klasse wird von der Informationsausgabe ausgeschlossen.

**E\_SPC\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_CLASS und wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_SPC\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_CLASS eine Liste von Benutzerkennungen enthält.

**E\_CLASS=\*NONE / array(16): <var: int: 1> / <integer 1..255>**

SPOOLOUT-Klassen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.



**E\_CLASS=\*NONE**

Keine SPOOLOUT-Klasse wird von der Informationsausgabe ausgeschlossen.

**E\_CLASS=array(16): <var: int: 1>**

Liste der SPOOLOUT-Klassen, über die keine Informationen ausgegeben werden sollen. E\_SPC\_T=\*LIST muss dafür gesetzt sein.

**E\_CLASS=<integer 1..255>**

SPOOLOUT-Klasse, die von der Informationsausgabe ausgeschlossen werden soll.

**E\_ACC\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für Abrechnungsnummern der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_ACC\_T=\*NONE**

Keine Abrechnungsnummer wird von der Informationsausgabe ausgeschlossen.

**E\_ACC\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_ACC wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_ACC\_T= \*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_ACC eine Liste von Benutzerkennungen enthält.

**E\_ACC=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Abrechnungsnummern der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden sollen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_ACC=\*NONE**

Keine Abrechnungsnummer wird von der Informationsausgabe ausgeschlossen.

**E\_ACC=array(16): <var: char: 8>**

Liste der Abrechnungsnummern, über die keine Informationen ausgegeben werden sollen. E\_ACC\_T=\*LIST muss dafür gesetzt sein.

**E\_ACC=<c-string 1..8: c-string 1-8>**

Abrechnungsnummer, die von der Informationsausgabe ausgeschlossen werden soll.

**E\_DESTY=\*NONE / list-poss(4): \*NONE / \*DEVICE / \*CENTRAL**

Druckaufträge, die auf den angegebenen Ausgabemedien (Pool oder Geräte) verarbeitet werden sollen, sind von der Informationsausgabe ausgeschlossen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DESTY=\*NONE**

da kein Ausgabemedium angegeben ist, wird kein Druckauftrag von der Informationsausgabe ausgeschlossen.

**E\_DESTY=\*DEVICE**

Informationen über Druckaufträge für die in E\_DEV\_T, E\_DVNAM und E\_DVNAW angegebenen Geräte werden von der Ausgabe ausgeschlossen.

**E\_DESTY=\*CENTRAL**

Informationen über Druckaufträge, die auf SPOOL-Geräten des zentralen Printer-Pools verarbeitet werden sollen, werden von der Informationsausgabe ausgeschlossen.

**E\_DEV\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für Namen der Geräte fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DEV\_T=\*NONE**

Kein Gerät wird von der Informationsausgabe ausgeschlossen.

**E\_DEV\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_DVNAM und E\_DVNAW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_DEV\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_DVNAM eine Liste von Benutzerkennungen enthält.

**E\_DEV\_T=\*WILD**

Die Geräte werden mit Platzhaltern teilqualifiziert angegeben.

**E\_DVNAM=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Namen der Geräte, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DVNAM=\*NONE**

Kein Gerät wird von der Informationsausgabe ausgeschlossen.

**E\_DVNAM=array(16): <var: char: 8>**

Liste der Geräte, über die keine Informationen ausgegeben werden sollen.

E\_DEV\_T=\*LIST muss dafür gesetzt sein.

**E\_DVNAM=<c-string 1..8: c-string 1-8>**

Gerät, das von der Informationsausgabe ausgeschlossen werden soll.

**E\_DVNAW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Namen der Geräte, die von der Informationsausgabe ausgeschlossen werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DVNAW=\*NONE**

Kein Gerät wird von der Informationsausgabe ausgeschlossen.

**E\_DVNAW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Geräte werden mit Platzhaltern teilqualifiziert angegeben.

**E\_FRT\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Formatnamen der Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_FRT\_T=\*NONE**

Kein Formatname wird von der Informationsausgabe ausgeschlossen.

**E\_FRT\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_FRT\_N wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_FRT\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_FRT\_N eine Liste von Formatnamen enthält.

**E\_FRT\_N=\*NONE / array(16): <var: char : 63> / <c-string 1..63: c-string 1-63>**

Formatnamen der Druckaufträge, über die nicht informiert werden soll.

**E\_FRT\_N=\*NONE**

Kein Formatname wird von der Informationsausgabe ausgeschlossen.

**E\_FRT\_N=array(16): <var: char : 63>**

Liste der Formatnamen der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden soll. E\_FRT\_T=\*LIST muss dafür gesetzt sein.

**E\_FRT\_N=<c-string 1..63: c-string 1-63>**

Formatname der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden soll.

**E\_DES\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST / \*WILD**

Legt den Typ der Angabe für Namen der Geräte-Pools fest, die von der Informationsausgabe ausgeschlossen werden sollen.

Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DES\_T=\*NONE**

Kein Geräte-Pool wird von der Informationsausgabe ausgeschlossen.

**E\_DES\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_DSNAM und E\_DSNMW wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_DES\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_DSNAM eine Liste von Benutzerkennungen enthält.

**E\_DES\_T=\*WILD**

Die Geräte-Pools werden mit Platzhaltern teilqualifiziert angegeben.

**E\_DSNAM=\*NONE / array(16): <var: char: 8> / <c-string 1..8: c-string 1-8>**

Namen der Geräte-Pools, die von der Informationsausgabe ausgeschlossen werden sollen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DSNAM=\*NONE**

Kein Geräte-Pool wird von der Informationsausgabe ausgeschlossen.

**E\_DSNAM=array(16): <var: char: 8>**

Liste der Geräte-Pools, über die keine Informationen ausgegeben werden sollen. E\_DES\_T=\*LIST muss dafür gesetzt sein.

**E\_DSNAM=<c-string 1..8: c-string 1-8>**

Geräte-Pool, der von der Informationsausgabe ausgeschlossen werden soll.

**E\_DSNMW=\*NONE / <var: char: 24> / <c-string 1..24: c-string 1-24>**

Namen der Geräte-Pools, die von der Informationsausgabe ausgeschlossen werden sollen. Die Namen werden mit Platzhaltern teilqualifiziert angegeben. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_DSNMW=\*NONE**

Kein Geräte-Pool wird von der Informationsausgabe ausgeschlossen.

**E\_DSNMW=<var: char: 24> / <c-string 1..24: c-string 1-24>**

Die Geräte-Pools werden mit Platzhaltern teilqualifiziert angegeben.

**E\_VPR\_T=\*NONE / <var: enum-of \_flag\_set: 1> / \*LIST**

Legt den Typ der Angabe für die Namen der virtuellen Druckers jener Druckaufträge fest, die von der Informationsausgabe ausgeschlossen werden sollen. Bei Angabe von \*NONE wirkt der Operand nicht als Auswahlkriterium.

**E\_VPR\_T=\*NONE**

Kein Namen eines virtuellen Druckers wird von der Informationsausgabe ausgeschlossen.

**E\_VPR\_T=<var: enum-of \_flag\_set: 1>**

Die Vereinbarung über den Typ der Angabe für E\_VPR\_N wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Für die Beziehung zwischen Wert und dem gewünschten Typ siehe Tabelle auf [Seite 315](#).

**E\_VPR\_T=\*LIST**

Mit \*LIST legen Sie fest, dass der Operand E\_VPR\_N eine Liste von Namen virtueller Drucker enthält.

**E\_VPR\_N=\*NONE / array(16): <var: char : 8> / <c-string 1..8: c-string 1-8>**

Namen virtueller Drucker der Druckaufträge, über die nicht informiert werden soll.

**E\_VPR\_N=\*NONE**

Kein virtueller Drucker wird von der Informationsausgabe ausgeschlossen.

**E\_VPR\_N=array(16): <var: char : 8>**

Liste der virtuellen Drucker der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden soll. E\_VPR\_T=\*LIST muss dafür gesetzt sein.

**E\_VPR\_N=<c-string 1..8: c-string 1-8>**

Virtueller Drucker der Druckaufträge, die von der Informationsausgabe ausgeschlossen werden soll.

## Langform der Operandennamen

Kurzform im Makro	Langform (nicht im Makro anwenden)
S_SPN_T	select-SPOOLout-name-type
S_SPNAM	select-SPOOLout-name
S_SPN_W	select-SPOOLout-name-wildcard
S_USR_T	select-user-id-type
S_USER	select-user-identification
S_USERW	select-user-id-wildcard
S_HST_T	select-hostname-type
S_HOST	select-hostname
S_HOSTW	select-hostname_wildcard
S_SRV_T	select-server-name-type
S_SERV	select-server_name
S_SERVW	select-server_name_wildcard
S_FRM_T	select-form-name-type
(Teil 1 von 3)	
S_FORM	select-form-name
S_FORMW	select-form_name_wildcard
S_TSN_T	select-tsn-type
S_TSN	select-tsn
S_IDENT	select-ident
S_SRT_T	select-server_tsn-type
S_SVTSN	select-server_tsn
S_OVL_T	select-overlay-type
S_OVER	select-overlay
S_SPC_T	select-spoolout_class-type
S_CLASS	select-spoolout_class
S_ACC_T	select-account-type
S_ACC	select-account
S_FRT_T	select-format_name-type
S_FRT_N	select-format-name
S_DESTY	select-dest_type
S_SELTY	select-s_type
S_SPDVT	select-SPOOL-dev-type

<b>Kurzform im Makro</b>	<b>Langform (nicht im Makro anwenden)</b>
S_DEV_T	select-device-name-type
S_DVNAM	select-dev-name
S_DVNMW	select-dev_name_wildcard
S_DES_T	select-dest-name-type
S_DSNAM	select-dest-name
S_DSNMW	select-dest-name-wildcard
S_ERR_T	select-error-type
S-ERR-MSG	select-error-message
S-ERRMW	select-error-message_wildcard
S_FOB	select-fob
S_CSN	select-csn
S_PRIO	select-priority
S_ROT	select-rotation
S_JTYP	select-job-type
E_SPN_T	except-SPOOLout-name-type
E_SPNAM	except-SPOOLout-name
E_SPNAW	except-SPOOLout-name-wildcard
E_USR_T	except-user-id-type
E_USER	except-user-identification
E_USERW	except-user-id-wildcard
E_HST_T	except-hostname-type
E_HOST	except-hostname
E_HOSTW	except-hostname_wildcard
E_SRV_T	except-server-name-type
E_SERV	except-server_name
E_SERVW	except-server_name_wildcard
E_FRM_T	except-form-name-type
E_FORM	except-form_name
E_FORMW	except-form_name_wildcard
E_TSN_T	except-tsn-type specify
E_TSN	except-tsn
E_SRT_T	except-server-tsn-type

(Teil 2 von 3)

<b>Kurzform im Makro</b>	<b>Langform (nicht im Makro anwenden)</b>
E_SVTSN	except-server_tsn
E_OVL_T	except-overlay-type
E_OVER	except-overlay
E_SPC_T	except-SPOOLout-class-type
E_CLASS	except-spclass
E_ACC_T	except-account-type
E_ACC	except-account
E_DESTY	except-dest_type
E_DEV_T	except-device-name-type
E_DVNAM	except-dev-name
E_DVNAW	except-dev_name_wildcard
E_FRT_T	except-format_name-type
E_FRT_N	except-format-name
E_DES_T	except-dest-name-type
E_DSNAM	except-dest-name
E_DSNMW	except-dest-name-wildcard
S-ERR-T	select-error-message-type
S-ERMSG	select-error-message
S-ERRMW	select-error-message-wildcard
E_VPR_T	except-virtual_printer-type
E_VPR_N	except-virtual_printer-name

(Teil 3 von 3)



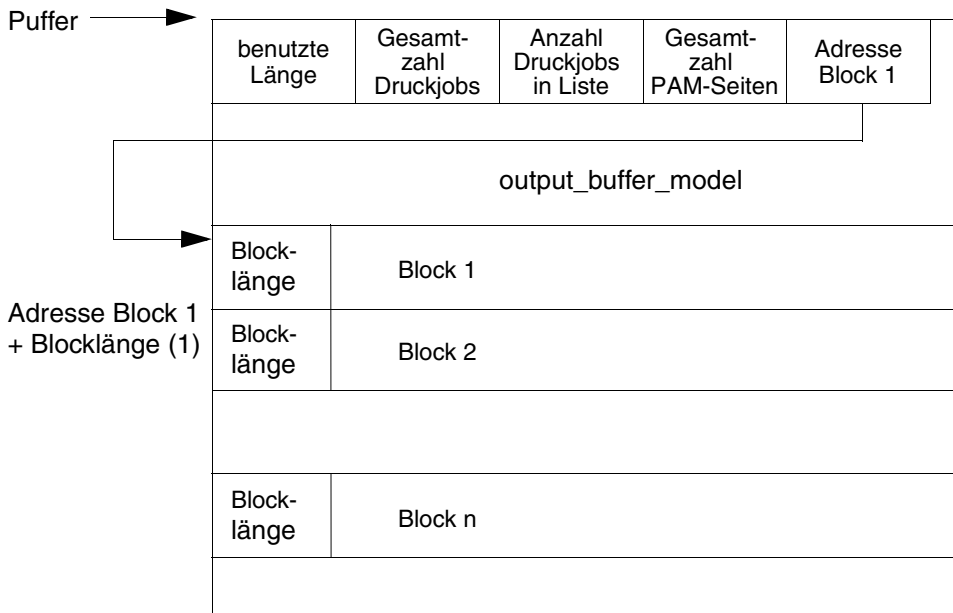
**Returncodes**

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	kein Fehler
02	00	0000	Druckauftrag nicht gefunden oder Verarbeitung nicht erlaubt
01	01	FFFF	Ungültige Parameter
02	01	FFFF	ungültige Adresse / Länge
03	01	FFFF	Ausgabebereich zu kurz
04	01	FFFF	Fehler während der Überprüfung der Parameter
00	20	FFFF	Systemfehler
02	40	FFFF	keine Autorisation
04	40	FFFF	Wertefehler im Operanden
05	40	FFFF	P/L-Version nicht unterstützt
00	80	FFFF	Subsystem nicht geladen
01	80	FFFF	SPOOL nicht geladen
02	80	FFFF	RSO nicht geladen
03	80	FFFF	PLAM nicht geladen
04	80	FFFF	JV nicht geladen
05	80	FFFF	SPS nicht geladen
06	80	FFFF	DPRINTCL nicht geladen
07	80	FFFF	DPRINTCM nicht geladen
08	80	FFFF	DPRINTSV nicht geladen
09	80	FFFF	DSEM nicht geladen

*Hinweise*

- Das Makro SHOPJST gibt diese Returncodes zurück. Zusätzlich können die Returncodes von HDRCHECK und \$VALID zurückgegeben werden. Wenn kein Datenbereich für den Header bereitgestellt wurde, wird der Wert X'0000003C' im Register 15 zurückgegeben.
- Die Information über die Druckaufträge wird in einem Ausgabepuffer abgelegt, der vom Benutzer bereitgestellt werden muss. Wenn der Puffer zu kurz definiert ist, wird der Aufruf des Makros mit dem Fehler X'03 01 FFFF' „Ausgabebereich zu kurz“ beendet. Um die komplette Information zu bekommen, muss ein Puffer mit zur Gesamtzahl der Druckaufträge passender Länge angefordert werden. Ein weiterer Aufruf von SHOPJAT bringt dann die gewünschte Information.

Der Abschnitt `_output_buffer_model` wird immer in den Puffer geschrieben. Er enthält die Gesamtzahl der Druckaufträge, die Anzahl der ausgewählten und beschriebenen Druckaufträge, die Adresse des ersten Blocks mit Information über einen Druckauftrag und die Länge dieses Blocks. Aus der Adresse des ersten Informationsblocks und dessen Länge lässt sich die Adresse eines jeden weiteren Informationsblocks errechnen.



- Informationen, die sich auf das Subsystem RSO beziehen, werden von RSO bereitgestellt, wenn dieses Subsystem geladen ist. Diese Informationen betreffen allerdings nur die von RSO gesteuerten Geräte.

### Beschreibung der Parameterliste

\* parameter list description

SSS_MDL		DS	OF	
SSSHDR	FHDR	MF=(C,SSS),EQUATES=NO		Standard header
* main return codes				
SSSRCNO		EQU	0	subcode1 = 00 : no error
SSSWARN		EQU	2	job not found or processing not permitted
*				
SSSPARM		EQU	1	subcode1 = 01 : wrong parameter
*				
SSSVAlD		EQU	2	invalid address/length
SSSSHRT		EQU	3	output area too short
SSSSYSE		EQU	0	subcode1 = 20 : system error
SSSPRIV		EQU	2	subcode1 = 40 : no authorization
*				
SSSVERR		EQU	4	operand value error
SSSVERE		EQU	5	p/l version not supported
SSSNOSS		EQU	0	subcode1 = 80 : subsystem not loaded
*				
SSSNSPO		EQU	1	spool not loaded
SSSNRSO		EQU	2	rso not loaded
SSSNPLM		EQU	3	plam not loaded
SSSNJV		EQU	4	jv not loaded
SSSNSPS		EQU	5	sps not loaded
SSSNDCL		EQU	6	dprintc1 not loaded
SSSNDCM		EQU	7	dprintcm not loaded
SSSNDSV		EQU	8	dprintsv not loaded
SSSNDSM		EQU	9	dsem not loaded
.001001	ANOP			
*				
SSRSOP		DS	A	address of rso_pl
SSRES1		DS	A	address of sps_pl
SSRES2		DS	A	address of dprint_pl
SSRES3		DS	A	address of dsem_pl
SSRES5		DS	A	
SSRES6		DS	A	
SSRES7		DS	A	
SSSCLUS		DS	CL8	This operand specifies the cluster in which the print-job is running. When a BS2000 cluster is specified, the TSN represents the gateway host in the remote specified cluster. Only a BS2000 cluster may be specified.
*				
*				
*				
*				
*				
*				
*				

SSSBUF@	DS	A	Address of a user-specified
*			area where the output must
*			be returned.
SSSBUFL	DS	F	user buffer length.
SSSINFO	DS	FL1	
* _info_set			
SSSORIG	EQU	1	info = *origin
SSSDEST	EQU	2	info = *destination
SSSTRTS	EQU	3	info = *traits
SSSDEVT	EQU	4	info = *device_type
SSSDVAL	EQU	5	info = *device_type
SSSSUMM	EQU	6	info = *summary
SSSDIST	EQU	7	info = *distributed
SSSSFLT	EQU	8	info = *destination
SSSRFLT	EQU	9	info = *destination
*			
SSSUN2	DS	XL3	
*			
SSSSEL	DS	0XL2640	select
*			
SSSASSPN	DS	0XL156	
SSSAFC8	DS	FL1	
* _flag_set			
SSSFALL	EQU	0	_type = *all
SSSFANY	EQU	0	_type = *any (ermsg)
SSSFNON	EQU	1	_type = *none
SSSFLOC	EQU	2	_type = *local
SSSFREM	EQU	3	_type = *remote
SSSFCEM	EQU	4	_type = *central
SSSFDEV	EQU	64	_type = *device
SSSFMDV	EQU	128	_type = *manag_dev
SSSFWLD	EQU	5	_type = *wildcard
SSSFLST	EQU	6	_type = *list
SSSFSTD	EQU	7	_type = *std
SSSFOWN	EQU	8	_type = *own
SSSFHOM	EQU	9	_type = *home
SSSFIDE	EQU	10	_type = *ident
SSSFALR	EQU	11	_type = *all ermsg
SSSANYF	EQU	11	_type = *any fob
SSSFONL	EQU	12	_type = *only
SSSFRNG	EQU	13	_type = *range
SSSFONE	EQU	14	_type = *one
SSSFNO	EQU	15	_type = *no
SSSFDES	EQU	16	_type=*destination
*			
SSSAWC8	DS	CL24	
SSSAUC8	DS	XL3	
SSSAEC8	DS	16CL8	

SSSAEC8#	EQU	16
*		
*		
SSSBSUID	DS	0XL156
SSSBFC8	DS	FL1
SSSBWC8	DS	CL24
SSSBUC8	DS	XL3
SSSBEC8	DS	16CL8
SSSBEC8#	EQU	16
*		
*		
SSSCSHOS	DS	0XL156
SSSCFC8	DS	FL1
SSSCWC8	DS	CL24
SSSCUC8	DS	XL3
SSSCFC8	DS	16CL8
SSSCFC8#	EQU	16
*		
*		
SSSDSSER	DS	0XL156
SSSDFC8	DS	FL1
SSSDWC8	DS	CL24
SSSDUC8	DS	XL3
SSSDEC8	DS	16CL8
SSSDEC8#	EQU	16
*		
**		
SSSEFOR	DS	0XL124
SSSEFC6	DS	FL1
SSSEWC6	DS	CL24
SSSEUC6	DS	XL3
SSSEEC6	DS	16CL6
SSSEEC6#	EQU	16
*		
*		
SSSFSTSN	DS	0XL92
SSSFFC4	DS	FL1
SSSFWC4	DS	CL24
SSSFUC4	DS	XL3
SSSFEC4	DS	16CL4
SSSFEC4#	EQU	16
*		
*		
SSSGSIDE	DS	0XL68
SSSGFI4	DS	FL1
SSSGUN1	DS	XL3
SSSGEI4	DS	16F
*		

*			
SSSHSSTS	DS	0XL92	
SSSHFC4	DS	FL1	
SSSHWC4	DS	CL24	
SSSHUC4	DS	XL3	
SSSHEC4	DS	16CL4	
SSSHEC4#	EQU	16	
*			
*			
SSSISOVE	DS	0XL60	
SSSIFC2	DS	FL1	
SSSIWC2	DS	CL24	
SSSIUC2	DS	XL3	
SSSIEC2	DS	16CL2	
SSSIEC2#	EQU	16	
*			
*			
SSSJSSPC	DS	0XL20	
SSSJFI1	DS	FL1	
SSSJUI2	DS	XL3	
SSSJEI1	DS	16X	
*			
*			
SSSKSACC	DS	0XL156	
SSSKFC8	DS	FL1	
SSSKWC8	DS	CL24	
SSSKUC8	DS	XL3	
SSSKEC8	DS	16CL8	
SSSKEC8#	EQU	16	
*			
*			
SSSQSFTN	DS	0XL1036	
SSSQFC63	DS	FL1	
SSSQWC63	DS	CL24	
SSSQUC63	DS	XL3	
SSSQEC63	DS	16CL63	
SSSQEC63#	EQU	16	
*			
*			
SSSLSDDES	DS	0XL169	
SSSLFLDS	DS	FL1	
SSSLSTDS	DS	FL1	
* sel_type_set			
SSSSTMA	EQU	128	sel-type = *may
SSSSTMU	EQU	64	sel-type = *must
*			
SSSLU1DS	DS	XL2	
SSSLDTDS	DS	FL1	

SSSLNADS	DS	CL24	
SSSLU2DS	DS	XL3	
SSSLDSEL	DS	16CL8	
SSSLDSEL#	EQU	16	
*			
SSSLDVDS	DS	0XL8	
SSSLDTB1	DS	AL1	device type byte1
SSSLLP	EQU	X'80'	LP printer
SSSLND	EQU	X'40'	ND printer
SSSLHP	EQU	X'20'	HP printer
SSSLLP48	EQU	X'10'	LP48 printer
SSSLLP65	EQU	X'08'	LP65 printer
SSSLTAPE	EQU	X'04'	TAPE
SSSLPEM	EQU	X'01'	LP-EMUL printer
SSSLDTB2	DS	AL1	device type byte2
SSSLDSU1	EQU	X'C0'	
SSSLHP90	EQU	X'20'	HP90 printer
SSSLAP50	EQU	X'10'	APA-2050 printer
SSSLAP90	EQU	X'08'	APA-2090 printer
SSSLTW90	EQU	X'04'	TWIN-2090 printer
SSSLDSU2	EQU	X'03'	
SSSLDTB3	DS	AL1	device type byte3
SSSLDSU3	EQU	X'FF'	
SSSLDTB4	DS	AL1	device type byte4
SSSLPCL	EQU	X'80'	PCL printer
SSSLDSU4	EQU	X'7F'	
SSSLDTB5	DS	AL1	device type byte5
SSSLDSU5	EQU	X'FF'	
SSSLDTB6	DS	AL1	device type byte6
SSSLDSU6	EQU	X'FF'	
SSSLDTB7	DS	AL1	device type byte7
SSSLDSU7	EQU	X'FF'	
SSSLDTB8	DS	AL1	device type byte8
SSSLDSU8	EQU	X'F0'	
SSSLVIRT	EQU	X'08'	VIRTUAL printer
SSSLDSU9	EQU	X'07'	
*			
SSSLUDS	DS	XL1	
*			
SSSUN4	DS	XL3	
*			
SSSMSFOB	DS	0XL8	
SSSMFR2	DS	FL1	
SSSMUN1	DS	XL1	
SSSMLR2	DS	H	
SSSMHR2	DS	H	
SSSMUR2	DS	XL2	
*			

*			
SSSNCSN	DS	0XL8	
SSSNFR2	DS	FL1	
SSSNUN1	DS	XL1	
SSSNLR2	DS	H	
SSSNHR2	DS	H	
SSSNUR2	DS	XL2	
*			
*			
SSSOSPRI	DS	0XL4	
SSSOFR1	DS	FL1	
SSSOLR1	DS	X	
SSS0HR1	DS	X	
SSSOUR1	DS	XL1	
*			
SSSSROT	DS	FL1	
* rotation_set			
SSSR0TA	EQU	0	rotation = *any
SSSR0TY	EQU	1	rotation = *yes
SSSR0TN	EQU	2	rotation = *no
*			
SSSSJTY	DS	16X	
*			
SSRSVPN	DS	0XL156	
SSSRFC8	DS	FL1	
SSSRWC8	DS	CL24	
SSSRUC8	DS	XL3	
SSSREC8	DS	16CL8	
SSSREC8#	EQU	16	
*			
SSSUN5	DS	XL3	
*			
*			
SSSEXC	DS	0XL2529	_except
*			
SSS0ESPN	DS	0XL156	
SSS0FC8	DS	FL1	
SSS0WC8	DS	CL24	
SSS0UC8	DS	XL3	
SSS0EC8	DS	16CL8	
SSS0EC8#	EQU	16	
*			
*			
SSS1EUID	DS	0XL156	
SSS1FC8	DS	FL1	
SSS1WC8	DS	CL24	
SSS1UC8	DS	XL3	
SSS1EC8	DS	16CL8	



SSS1EC8#	EQU	16
*		
*		
SSS2EH0S	DS	0XL156
SSS2FC8	DS	FL1
SSS2WC8	DS	CL24
SSS2UC8	DS	XL3
SSS2EC8	DS	16CL8
SSS2EC8#	EQU	16
*		
*		
SSS3ESER	DS	0XL156
SSS3FC8	DS	FL1
SSS3WC8	DS	CL24
SSS3UC8	DS	XL3
SSS3EC8	DS	16CL8
SSS3EC8#	EQU	16
*		
*		
+SSS4EF0R	DS	0XL124
SSS4FC6	DS	FL1
SSS4WC6	DS	CL24
SSS4UC6	DS	XL3
SSS4EC6	DS	16CL6
SSS4EC6#	EQU	16
*		
*		
SSS5ETSN	DS	0XL92
SSS5FC4	DS	FL1
SSS5WC4	DS	CL24
SSS5UC4	DS	XL3
SSS5EC4	DS	16CL4
SSS5EC4#	EQU	16
*		
*		
SSS6ESTS	DS	0XL92
SSS6FC4	DS	FL1
SSS6WC4	DS	CL24
SSS6UC4	DS	XL3
SSS6EC4	DS	16CL4
SSS6EC4#	EQU	16
*		
*		
SSS7E0VE	DS	0XL60
SSS7FC2	DS	FL1
SSS7WC2	DS	CL24
SSS7UC2	DS	XL3
SSS7EC2	DS	16CL2

SSS7EC2#	EQU	16	
*			
*			
SSS8ESPC	DS	0XL20	
SSS8FI1	DS	FL1	
SSS8UI2	DS	XL3	
SSS8EI1	DS	16X	
*			
*			
SSS9EACC	DS	0XL156	
SSS9FC8	DS	FL1	
SSS9WC8	DS	CL24	
SSS9UC8	DS	XL3	
SSS9EC8	DS	16CL8	
SSS9EC8#	EQU	16	
*			
*			
SSSSFTN	DS	0XL1036	
SSSSFC63	DS	FL1	
SSSSWC63	DS	CL24	
SSSSUC63	DS	XL3	
SSSSEC63	DS	16CL63	
SSSSEC63#	EQU	16	
*			
*			
SSSZEDES	DS	0XL169	
SSSZFLDS	DS	FL1	
SSSZSTDS	DS	FL1	
SSSZU1DS	DS	XL2	
SSSZDTDS	DS	FL1	
SSSZNADS	DS	CL24	
SSSZU2DS	DS	XL3	
SSSZDSEL	DS	16CL8	
SSSZDSEL#	EQU	16	
*			
SSSZDVDS	DS	0XL8	
+SSSZDTB1	DS	AL1	device type byte1
SSSZLP	EQU	X'80'	LP printer
SSSZND	EQU	X'40'	ND printer
SSSZHP	EQU	X'20'	HP printer
SSSZLP48	EQU	X'10'	LP48 printer
SSSZLP65	EQU	X'08'	LP65 printer
SSSZTAPE	EQU	X'04'	TAPE
SSSZLPEM	EQU	X'01'	LP-EMUL printer
SSSZDTB2	DS	AL1	device type byte2
SSSZDSU1	EQU	X'CO'	
SSSZHP90	EQU	X'20'	HP90 printer
SSSZAP50	EQU	X'10'	APA-2050 printer

```

SSSZAP90      EQU  X'08'      APA-2090 printer
SSSZTW90      EQU  X'04'      TWIN-2090 printer
SSSZDSU2      EQU  X'03'
SSSZDTB3      DS    AL1      device type byte3
SSSZDSU3      EQU  X'FF'
SSSZDTB4      DS    AL1      device type byte4
SSSZPCL       EQU  X'80'      PCL printer
SSSZDSU4      EQU  X'7F'
SSSZDTB5      DS    AL1      device type byte5
SSSZDSU5      EQU  X'FF'
SSSZDTB6      DS    AL1      device type byte6
SSSZDSU6      EQU  X'FF'
SSSZDTB7      DS    AL1      device type byte7
SSSZDSU7      EQU  X'FF'
SSSZDTB8      DS    AL1      device type byte8
SSSZDSU8      EQU  X'F0'
SSSZVIRT      EQU  X'08'      VIRTUAL printer
SSSZDSU9      EQU  X'07'
*
SSSZUDS       DS    XL1
*
*
SSSTVVPN      DS    0XL156
SSSTFC8       DS    FL1
SSSTWC8       DS    CL24
SSSTUC8       DS    XL3
SSSTEC8       DS    16CL8
SSSTEC8#      EQU  16
*
*
*
SSSMSERR      DS    0XL156
SSSMFC8       DS    FL1
SSSMWC8       DS    CL24
SSSMUC8       DS    XL3
SSSMEC8       DS    16CL8
SSSMEC8#      EQU  16
*
SSSUN9        DS    XL11
SSS#          EQU  *-SSSHDR

```

## 4.1 Unterstützende Makros für MODPJAT

Die folgenden Makros unterstützen den Einsatz des Makros MODPJAT. Mit diesen Makros werden subsystemspezifische Parameter für das Makro MODPJAT gesetzt.

„[MODJRSO - RSO-Parameter für MODPJAT-Makro festlegen](#)“ auf Seite 357

Mit diesem Makro werden Parameter für das Subsystem RSO gesetzt.

„[MPJADPC - Dprint-Parameter für MODPJAT-Makro definieren](#)“ auf Seite 364

Mit diesem Makro werden Parameter für das Subsystem Dprint gesetzt.

„[MPJASPS - SPS-Parameter für MODPJAT-Makro definieren](#)“ auf Seite 366

Mit diesem Makro werden Parameter für das Subsystem SPS gesetzt.

## MODJRSO - RSO-Parameter für MODPJAT-Makro festlegen

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Der Makro MODJRSO hinterlegt RSO-spezifische Werte für den Aufruf des Makros MODPJAT in einer Parameterliste.

### Format

Operation	Operanden
MODJRSO	<pre>,VARIANT=<u>001</u> / &lt;c-string 3..3&gt;  ,CHARATT=<u>*STD</u> / *ALL / *RESTRICTED / &lt;var: enum-of _char_att_type_set:1&gt;  ,REDIR=<u>*STD</u> / *YES / *NO / &lt;var: enum-of _redir_type_set:1&gt;  TOPOFF=<u>*UNCHANGED</u> / *IGNORE / &lt;integer -255..255&gt;  LEFTOFF=<u>*UNCHANGED</u> / *IGNORE / &lt;integer -255..255&gt;  URL=(<i>urlptr,urlen,prtype</i>)   <i>urlptr</i>: &lt;var: pointer&gt; / *NONE / *UNCHANGED   <i>urlen</i>: &lt;var: int 0..65535&gt;/ &lt;integer 1..1023&gt; / *NONE   <i>prtype</i>: &lt;var: enum-of:1 _PRTYPE_SET&gt; / *NONE /*UNCHANGED /     *4850_PCL / *4830_PCL / *4818_PCL / *9000_EPSQ /     *9000_EPLQ / *9000_EPFX / *9000_PRO / *9000_PS /     *9046 / *9045_ANSI / *9015 / *4825_PCL / *4822_PCL /     *9001 / *9025 / *9013 / *9000 / *9001_31 / *9011 / *9012 /     *9022 / *9014 / *9021 / *9022_200 / *9000_PCL /     *4824_PCL / *9026_RENO / *9026_PCL / *4812 /     *9097 / *4011 / *4813 / *4821_PCL / *2030_PCL</pre>

## Operandenbeschreibung

### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **CHARATT=\*STD / \*ALL / \*RESTRICTED / <var: enum-of \_char\_att\_type\_set:1>**

Legt fest, ob für den RSO-Auftrag alle oder nur bestimmte Zeichensatz-Eigenschaften berücksichtigt werden. Solche Eigenschaften sind z.B. Schriftart, Schönschrift (NLQ), Farbe u.s.w. (siehe Kommando oder SPERVE-Anweisung SHOW-SPOOL-CHARACTER-SETS). Für Deck- und Schlussblätter ist dieser Operand nicht wirksam.

### **CHARATT=\*STD**

Die als Standard voreingestellten Eigenschaften der zum Ausdrucken benutzten Zeichensätze werden für den aktuellen RSO-Auftrag berücksichtigt.

### **CHARATT=\*ALL**

Alle Eigenschaften der zum Ausdrucken benutzten Zeichensätze werden für den aktuellen RSO-Auftrag berücksichtigt.

### **CHARATT=\*RESTRICTED**

Nur die drei Eigenschaften Schriftart (CHARACTER TYPE), Sprache (LANGUAGE) und NLQ (NEAR-LETTER-QUALITY) werden berücksichtigt.

### **CHARATT=<var: enum-of \_char\_att\_type\_set:1>**

Die Vereinbarung bzgl. der zu verwendenden Zeichensatz-Eigenschaften wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und gewünschter Funktion:

0	*STD
1	*ALL
2	*RESTRICTED
siehe Beschreibung der Parameterliste ab <a href="#">Seite 362</a> .	

### **REDIR=\*STD / \*YES / \*NO / <var: enum-of \_redir\_type\_set:1>**

Gibt an, ob ein Geräteverwalter den RSO-Auftrag auf einen anderen Drucker umleiten darf. Das Umleiten von Aufträgen durch den Benutzer oder das Umleiten von Aufträgen durch die Systembetreuung wird dadurch nicht berührt.

### **REDIR=\*STD / \*YES**

Ein Geräteverwalter darf den RSO-Auftrag auf einen anderen Drucker umleiten.

### **REDIR=\*NO**

Ein Geräteverwalter darf den RSO-Auftrag nicht auf einen anderen Drucker umleiten.

**REDIR=<var: enum-of \_redir\_type\_set:1>**

Die Vereinbarung bzgl. des Umleitens des RSO-Auftrags wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und gewünschter Funktion:

0	*STD
1	*YES
2	*NO
siehe Beschreibung der Parameterliste ab <a href="#">Seite 362</a> .	

**TOPOFF=\*UNCHANGED / \*IGNORE / <integer -255..255>**

Legt den Abstand des oberen Randes der Druckseite vom oberen Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der der Druckseite nicht beachtet wird.

Dieser Operand ist nur für die Drucker 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 9000-PCL, 9021, 9022-200 und 2030-PCL zulässig.

*Hinweis*

Mit DOCFOR=\*SPECIAL\_FORMAT zusammen kann nur TOPOFF=\*IGNORE angewandt werden.

**TOPOFF=\*IGNORE**

Die Druckersteuerung setzt keinen Abstand zwischen der Druckseite und dem oberen Seitenrand fest. Die Druckervoreinstellung oder die Einstellung durch die Prologdatei legen den Abstand fest, der beim Ausdrucken des Dokuments wirksam wird.

**TOPOFF=<integer -255..255>**

Der zulässige Wertebereich ist -255 bis +255. Positive Werte verschieben die Druckseite nach unten. Negative Werte verschieben die Druckseite nach oben.

**LEFTOFF=\*UNCHANGED / \*IGNORE/ <integer -255..255>**

Legt den Abstand des linken Randes der Druckseite vom linken Rand der physikalischen Seite in Millimetern fest. Zuerst wird die Druckseite auf dem Papier positioniert, und erst danach die Schrift innerhalb der Druckseite gedreht und positioniert. Das bedeutet, dass beim Verschieben der Druckseite gegenüber der Papierseite die Orientierung des Textes innerhalb der der Druckseite nicht beachtet wird.

Dieser Operand ist nur für die Drucker 4818-PCL, 4821-PCL, 4822-PCL, 4824-PCL, 4825-PCL, 9000-PCL, 9021, 9022-200, 9026-PCL und 20930-PCL zulässig.

*Hinweis*

Mit DOCUMENT\_FORMAT=\*SPECIAL\_FORMAT zusammen kann nur LEFTOFF=\*IGNORE angewandt werden.

**LEFTOFF=\*IGNORE**

Die Druckersteuerung setzt keinen Abstand zwischen der Druckseite und dem linken Seitenrand fest. Die Druckervoreinstellung oder die Einstellung durch die Prologdatei legen den Abstand fest, der beim Ausdrucken des Dokuments wirksam wird.

**LEFTOFF=<integer -255..255>**

Der zulässige Wertebereich ist -255 bis +255. Positive Werte verschieben die Druckseite nach rechts. Negative Werte verschieben die Druckseite nach links.

**URL=(*urlptr,urlen,prtype*)**

Bezeichnet den IPP-Drucker, der durch seine URL adressiert wird.

***urlptr*: <var: pointer>**

Bereich, der die Drucker-URL enthält

***urlptr*: \*NONE**

Es wurde keine Drucker-URL angegeben.

***urlptr*: \*UNCHANGED**

Standard-Einstellung. Die aktuelle Einstellung wird nicht geändert.

***urlen*: <var: int 0..65535>**

Länge des Bereichs.

***urlen*: <integer 1..1023>**

Länge des Bereichs.

***urlen*: \*NONE**

Standard-Einstellung. Es wurde keine URL angegeben.

***prtype*: <var: enum-of:1 \_PRTYPE\_SET>**

Der Druckertyp wird nicht direkt durch einen Operandenwert angegeben. Stattdessen wird er indirekt durch ein Feld mit konstantem Inhalt (Equate) angegeben. Es kann ein Integer-Wert in der Konstanten oder im zugehörigen Feld gespeichert werden. Folgende Beziehung besteht zwischen den Werten und den gewünschten Funktionen:

Wert	Druckertyp
72	*4850_PCL
73	*4830_PCL
74	*4818_PCL
75	*9000_EPSQ
76	*9000_EPLQ
77	*9000_EPFX



<b>Wert</b>	<b>Druckertyp</b>
78	*9000_PRO
79	*9000_PS
7B	*9046
7D	*9045_ANSI
7E	*9015
7F	*4825_PCL
80	*4822_PCL
84	*9001
88	*9025
8A	*9013
8B	*9000
8C	*9001_31
8D	*9011
8E	*9012
8F	*9022
A0	*9014
A1	*9021
A2	*9022_200
A4	*9000_PCL
A6	*4824_PCL
A7	*9026_RENO
A8	*9026_PCL
A9	*4812
AA	*9097
AB	*4011
AC	*4813
AD	*4821_PCL
AE	*2030_PCL

***prtype: \*NONE***

Es wurde keine URL angegeben.

***prtype: \*UNCHANGED***

Standard-Einstellung. Die aktuelle Einstellung wird nicht geändert.

***prtype: \*4850\_PCL / \*4830\_PCL / \*4818\_PCL / \*9000\_EPSQ / \*9000\_EPLQ / \*9000\_EPFX / \*9000\_PRO / \*9000\_PS / \*9046 / \*9045\_ANSI / \*9015 / \*4825\_PCL / \*4822\_PCL / \*9001 / \*9025 / \*9013 / \*9000 / \*9001\_31 / \*9011 / \*9012 / \*9022 / \*9014 / \*9021 / \*9022\_200 / \*9000\_PCL / \*4824\_PCL / \*9026\_RENO / \*9026\_PCL / \*4812 / \*9097 / \*4011 / \*4813 / \*4821\_PCL / \*2030\_PCL***

RSO-Gerätetyp des Zieldruckers.

## Beschreibung der Parameterliste

\* parameter list description

SROAMDLS DS OF

SROAHDR FHDR MF=(C,SROA),EQUATES=NO

Standard header

SROACHRA DS FL1

Specifies whether all or only specific character set attributes are to be used for the print-job. These attributes include character type, near letter quality, color, etc. This operand does not apply to header and trailer pages.

\*

\*

\*

\*

\*

\*

\*

\*

\*

\* \_char\_att\_type\_set

AIF ('&EQUATES' NE 'YES').Q01001

SROACSTD EQU 0

SROACALL EQU 1

SROACRST EQU 2

.Q01001 ANOP

\*

SROARDIR DS FL1

Specifies whether a device administrator may redirect print-job to another printer. Redirection of print-job by users or system-administration is not affected by this operand.

\*

\*

\*

\*

\*

\*

\*

\* \_redir\_type\_set

AIF ('&EQUATES' NE 'YES').Q01002

SROARSTD EQU 0

SROARYES EQU 1

SROARNO EQU 2

.Q01002 ANOP

```

*
SROASPEC DS AL1 specified; for compatibility
* with MODJRSO
      AIF ('&EQUATES' NE 'YES').Q01003
SROACSPC EQU X'80'
SROARSPC EQU X'40'
SROATSPC EQU X'20'
SROALSPC EQU X'10'
SROAUNUS EQU X'0F'
.Q01003 ANOP
SROAFIL1 DS CL1 slack byte
SROATOPO DS H Specifies the length of the
* white band that must be
* inserted
SROALEFO DS H Specifies the width of the
* white band that must be
* inserted
SROAUNS1 DS XL8 unused
      AIF ('&EQUATES' NE 'YES').Q01004
SROA# EQU *-SROAHDR

```

## MPJADPC - Dprint-Parameter für MODPJAT-Makro definieren

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Mit diesem Makro werden die Dprint-Parameter für den MODPJAT-Makro definiert.

### Format

Operation	Operanden
MPJADPC	VARIANT= <u>001</u> / <c-string 3..3>  ,FORMAT= <u>*STD</u> / <var: char: 63> / <c-string 1..63>  .RESLOC= <u>*STD</u> / *HOME / *SERVER / <var: enum-of_resloc_set:1>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**FORMAT=\*STD / <var: char: 63> / <c-string 1..63>**

Art des Dokument-Inhalts.

**FORMAT=\*STD**

Standardmäßig wird keine spezielle Formatbezeichnung angenommen.

**FORMAT=<var: char: 63>**

Name des Feldes mit der Formatbezeichnung.

**FORMAT=<c-string 1..63>**

Formatbezeichnung; Direktangabe.

**RESLOC=\*STD / \*HOME / \*SERVER / <var: enum-of\_resloc\_set:1>**

Lage der Ressourcen, die für den Druck verwendet werden.

**RESLOC=\*STD**

Standardmäßig werden die Werte aus der SPOOL-Parameterdatei angenommen.

**RESLOC=\*HOME**

Die Client-Ressourcen werden verwendet.

**RESLOC=\*SERVER**

Die Server-Ressourcen werden verwendet.

**RESLOC=<var: enum-of\_resloc\_set:1>**

Name des Equates für den Operandenwert. Nur in Verbindung mit MF=M möglich.

**Beschreibung der Parameterliste**

```

*   parameter list description
DMAPL   DS    0F
DMAHDR  FHDR  MF=(C,DMA),EQUATES=NO   Standard header
DMAUNU1 DS    XL8                       reserved for compatibility
*                                           with print command
DMAFMT  DS    CL63                       document content type
DMARESL DS    FL1                       location of the resources
*                                           used to process the
*                                           print-job is submitted
*   _resloc_set
      AIF  ('&EQUATES' NE 'YES').Q01001
DMARSTD EQU    0                       resource-location = *std
DMARHOM EQU    1                       resource-location = *home
DMARSRV EQU    2                       resource-location = *server
.Q01001 ANOP
*
DMASP1  DS    AL1                       specified
      AIF  ('&EQUATES' NE 'YES').Q01002
DMASP11 EQU  X'80'
DMASPUN1 EQU  X'40'
*                                           reserved for compatibility
*                                           with print command
DMASP13 EQU  X'20'
DMASP14 EQU  X'10'
DMASP1F EQU  X'0F'
.Q01002 ANOP
DMAUNU2 DS    XL63                       reserved for compatibility
*                                           with print command
DMAFIL1 DS    CL4                       slack bytes
      AIF  ('&EQUATES' NE 'YES').Q01003
DMA#    EQU    *-DMAHDR
.Q01003 ANOP

```

## MPJASPS - SPS-Parameter für MODPJAT-Makro definieren

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Mit diesem Makro werden die SPS-Parameter für den MODPJAT-Makro definiert.

### Format

Operation	Operanden
MPJASPS	VARIANT= <u>001</u> / <c-string 3..3>  FONTS= <u>*UNCHANGED</u> / array(4): <var: char: 8: alphanum_name_1..8>  PAGEDEF= <u>*UNCHANGED</u> / <var: char: 8: alphanum_name_1..8>  FORMDEF= <u>*UNCHANGED</u> / <var: char: 8: alphanum_name_1..8>  TRC= <u>*UNCHANGED</u> / <var: enum-of _trc_set: 1> / *YES / *NO  MSGPAGE= <u>*UNCHANGED</u> / <var: enum-of _msgpage_set: 1> / *YES / *NO  FRONTOVL= <u>*UNCHANGED</u> / <var: char: 8:alphanum_name_1..8> / *STD  BACKOVL= <u>*UNCHANGED</u> / <var: char: 8:alphanum_name_1..8> / *STD

## Operandenbeschreibung

### **VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### **FONTS=\*UNCHANGED / array(4): <var: char: 8: alphanum\_name\_1..8>**

Vereinbart die Fonts, die bei Verwendung des Operanden TABLE-REFERENCE-CHAR der Kommandos PRINT-DOCUMENT und MODIFY-PRINT-JOB-ATTRIBUTES beziehungsweise des Operanden TRC der Makros PRNTDOC und MPJASPS von SPS auszuwerten sind.

### **FONTS=\*UNCHANGED**

Die beim Erteilen des Druckauftrags vereinbarte Einstellung bleibt unverändert erhalten.

### **FONTS=array(4): <var: char: 8: alphanum\_name\_1..8>**

Name der Fonts, die mit dem TRC-Wert in der Druckdatei (der jeweils einen bestimmten Font repräsentiert) zu verknüpfen sind. Unabhängig von der Anzahl unterschiedlicher TRC-Werte in der Datei können maximal vier Fonts, die in Form einer Liste anzugeben sind, für den Ausdruck der Datensätze verwendet werden. Höhere TRC-Werte als X'03' (entspreche dem vierten Element der Liste) verweisen dabei automatisch auf den in der Liste erstgenannten Font.

### **PAGEDEF=\*UNCHANGED / <var: char: 8: alphanum\_name\_1..8>**

Gibt an, welche Seiten-Definition für den Ausdruck auf APA-Druckern benutzt werden soll.

### **PAGEDEF=\*UNCHANGED**

Die beim Erteilen des Druckauftrags vereinbarte Einstellung bleibt unverändert erhalten.

### **PAGEDEF=<var: char: 8: alphanum\_name\_1..8>**

Nur für APA-Drucker:

Die Seiten-Definition mit dem angegebenen Namen soll benutzt werden. Diese muss sich in der SPSLIB befinden.



Die ersten beiden Zeichen des angegebenen Namens müssen „P1“ sein. Andernfalls wird die Änderung des Druckauftrags abgewiesen.

### **FORMDEF=\*UNCHANGED / <var: char: 8: alphanum\_name\_1..8>**

Gibt an, welche Format-Definition für den Ausdruck auf APA-Druckern benutzt werden soll.

### **FORMDEF=\*UNCHANGED**

Die beim Erteilen des Druckauftrags vereinbarte Einstellung bleibt unverändert erhalten.

### **FORMDEF=<var: char: 8: alphanum\_name\_1..8>**

Die Format-Definition mit dem angegebenen Namen soll benutzt werden. Diese muss sich in der SPSLIB befinden.



Die ersten beiden Zeichen des angegebenen Namens müssen „F1“ sein. Andernfalls wird das Kommando abgewiesen.

**TRC=\*UNCHANGED / <var: enum-of \_trc\_set:1> / \*YES / \*NO**

Legt fest, ob der Anwender die Auswahl von Fonts zur Druckseitengestaltung über Steuerzeichen im Text vornimmt. Diese Auswahl kann mit Hilfe der TRC-Steuerzeichen (Table Reference-Character) vorgenommen werden, die innerhalb der Datensätze auf eine Liste von Fonts verweisen.

Die Liste kann entweder Bestandteil der Seitendefinition PAGEDEF sein oder mit Hilfe des Operanden FONTS bekannt gegeben werden. Die ausführliche Erläuterung zur Anwendung der TRC ist im Handbuch „[SPOOL \(BS2000/OSD\)](#)“ zu finden.

**TRC=\*UNCHANGED**

Die beim Erteilen des Druckauftrags vereinbarte Einstellung bleibt unverändert erhalten.

**TRC=<var: enum-of \_trc\_set:1>**

Die Vereinbarung, ob TRC-Steuerzeichen verwendet werden, wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und gewünschter Funktion:

1	*YES
2	*NO
4	*UNCHANGED

**TRC=\*YES**

Innerhalb der Druckdatei befinden sich Font-Auswahl-Steuerzeichen (TRC), die von SPS auszuwerten sind. Ein TRC-Wert repräsentiert jeweils einen bestimmten Font, mit dem der betreffende Datensatz auszudrucken ist. Die einzelnen Werte (von X'00' bis X'0F') sind entsprechend ihrer Reihenfolge mit den Einträgen in der FONTLIST (Fontliste der Seitendefinition PAGEDEF) zu verknüpfen oder sind entsprechend ihrer Reihenfolge explizit mit im Operanden CHARACTER-SETS angegebenen Fonts zu verknüpfen. Unabhängig von der Anzahl unterschiedlicher TRC-Werte in der Datei können maximal vier Fonts, die in Form einer Liste anzugeben sind, für den Ausdruck der Datensätze verwendet werden. Höhere TRC-Werte als X'03' (entspräche dem vierten Element der Liste) verweisen dabei automatisch auf den in der Liste erstgenannten Font.

**TRC=\*NO**

Innerhalb der Druckdatei sind keine Font-Auswahl-Steuerzeichen (TRC) hinterlegt oder diese sollen nicht ausgewertet werden.

**MSGPAGE=\*UNCHANGED / <var: enum-of \_msgpage\_set:1> / \*YES / \*NO**

Gibt an, ob die APA-Meldungsseite gedruckt werden soll oder nicht. Die APA-Meldungsseite enthält Fehlermeldungen und Warnungen. Falls kein APA-Drucker benutzt wird, wird die Angabe ignoriert.



**MSGPAGE=\*UNCHANGED**

Die beim Erteilen des Druckauftrags vereinbarte Einstellung bleibt unverändert erhalten.

**MSGPAGE=<var: enum-of \_msgpage\_set:1>**

Die Vereinbarung, ob die APA-Meldungsseite gedruckt wird, wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld mit konstantem Inhalt (EQUATE) angegeben. In der Konstanten bzw. dem entsprechenden Feld kann ein ganzzahliger Wert hinterlegt werden. Hier besteht folgende Beziehung zwischen Wert und gewünschter Funktion:

1	*YES
2	*NO
4	*UNCHANGED

**MSGPAGE=\*YES**

Die APA-Meldungsseite wird gedruckt. Falls kein APA-Drucker benutzt wird, wird die Angabe ignoriert.

**MSGPAGE=\*NO**

Die APA-Meldungsseite wird nicht gedruckt.

**FRONTOVL=\*UNCHANGED / <var: char: 8:alphanum\_name\_1..8> / \*STD**

Vereinbart, ob ein APA-Overlay für die Vorderseite verwendet werden soll.

**FRONTOVL=\*UNCHANGED**

Die aktuelle Einstellung wird nicht verändert.

**FRONTOVL=<var: char: 8:alphanum\_name\_1..8>**

Name des APA-Overlays, das für die Vorderseite verwendet werden soll.

**FRONTOVL=\*STD**

Es werden keine APA-Overlays für die Ausgabe benutzt, außer wenn in der zugehörigen Form Definition ein APA-Overlay angegeben ist.

**BACKOVL=\*UNCHANGED / <var: char: 8:alphanum\_name\_1..8> / \*STD**

Vereinbart, ob ein APA-Overlay für die Rückseite verwendet werden soll.

**BACKOVL=\*UNCHANGED**

Die aktuelle Einstellung wird nicht verändert.

**BACKOVL=<var: char: 8:alphanum\_name\_1..8>**

Name des APA-Overlays, das für die Rückseite verwendet werden soll.

**BACKOVL=\*STD**

Es werden keine APA-Overlays für die Ausgabe benutzt, außer wenn in der zugehörigen Form Definition ein APA-Overlay angegeben ist.

**Beschreibung der Parameterliste**

\* parameter list description

SPSMPL	DS	0F	
SPSMHDR	FHDR	MF=(C,SPSM),EQUATES=NO	Standard header
SPSMFNT	DS	4CL8	When TRCs are used, allows to reference a list of fonts
*			
SPSMFNT#	EQU	4	
SPSMFNT#	DS	CL8	Element name of the page definition to be used.
*			
SPSMFRMD	DS	CL8	Element name of the format definition to be used.
*			
SPSMTRC	DS	FL1	Specifies whether the file to print contains TRCs.
*			
*_trc_set			
SPSMTRCY	EQU	1	
SPSMTRCN	EQU	2	
SPSMTRCS	EQU	4	
*			
SPSMMSGP	DS	FL1	
*_msgpage_set			
SPSMMSGY	EQU	1	
SPSMMSGN	EQU	2	
SPSMMSGS	EQU	4	
*			
SPSMSP1	DS	AL1	specified
SPSMSP11	EQU	X'80'	
SPSMSP12	EQU	X'40'	
SPSMSP13	EQU	X'20'	
SPSMSP14	EQU	X'10'	
SPSMSP15	EQU	X'08'	
SPSMSP1F	EQU	X'07'	
SPSMFIL1	DS	CL1	slack byte
SPSM#	EQU	*-SPSMHDR	

## 4.2 Ausgabestrukturen des SHOPJAT-Makros

Die folgenden Makros beschreiben die Strukturen, mit denen der Ausgabepuffer des Makros SHOPJAT nach einer Statusabfrage gefüllt wird.

„[SJAODOF - Ausgabestruktur für INFO=\\*DOCUMENT\\_FORMAT“ auf Seite 372](#)

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJAT nach einer Abfrage mit INFO=\*DOCUMENT\_FORMAT gefüllt wird.

„[SJAODOP - Ausgabestruktur für INFO=\\*DOCUMENT\\_PART“ auf Seite 374](#)

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJAT nach einer Abfrage mit INFO=\*DOCUMENT\_PART gefüllt wird.

„[SJAOLCT - Ausgabestruktur für INFO=\\*LAYOUT\\_CONTROL“ auf Seite 377](#)

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJAT nach einer Abfrage mit INFO=\*LAYOUT\_CONTROL gefüllt wird.

„[SJAOPJC - Ausgabestruktur für INFO=\\*PRINT\\_JOB\\_CONTROL“ auf Seite 380](#)

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJAT nach einer Abfrage mit INFO=\*PRINT\_JOB\_CONTROL gefüllt wird.

„[SJAORSD - Ausgabestruktur für INFO=\\*RESOURCES\\_DESCRIPTION“ auf Seite 381](#)

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJAT nach einer Abfrage mit INFO=\*RESOURCES\_DESCRIPTION gefüllt wird.

„[SJAOTOP - Ausgabestruktur für INFO=\\*TO\\_PRINTER“ auf Seite 383](#)

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJAT nach einer Abfrage mit INFO=\*TO\_PRINTER gefüllt wird.

„[SPJARSO - RSO-Ausgabestruktur für SHOPJAT-Makro“ auf Seite 384](#)

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST ergänzt wird, wenn eine Abfrage einen mit RSO abzuwickelnden Druckauftrag betraf.

„[SPJADPC - Dprint-Ausgabestruktur für SHOPJAT-Makro“ auf Seite 385](#)

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST ergänzt wird, wenn eine Abfrage einen mit Dprint abzuwickelnden Druckauftrag betraf.

„[SPJASPS - SPS-Ausgabestruktur für SHOPJAT-Makro“ auf Seite 386](#)

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST ergänzt wird, wenn eine Abfrage einen mit SPS abzuwickelnden Druckauftrag betraf.

## SJAODOF - Ausgabestruktur für INFO=\*DOCUMENT\_FORMAT

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Informationen über die unter DOCUMENT\_FORMAT angesprochenen Optionen vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SJAODOF	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _doc_format_block_md1
SAFDPBM      DS    0H
SAFDFDF      DS    FL1          document format
*  _doc_format_set
SAFDFTE      EQU   0           text
SAFDFPG      EQU   1           page
SAFDFSP      EQU   2           special
*
*
SAFCONTROL_MODE DS    0XL3      _control_mode
SAFDOFO      DS    FL1
*  _contmod_type_set
SAFCTMP      EQU   0           page mode
SAFCTNO      EQU   1           no
SAFCTPH      EQU   2           physical
SAFCTML      EQU   4           logical
SAFCTLI      EQU   16          line mode
SAFCTTR      EQU   32          transparent
SAFCTMA      EQU   64          apa
*
SAFPCC       DS    FL1
```

```

*   _contmod_pcc_set
SAFTOPY      EQU 0      yes
SAFTOPN      EQU 1      no
*
SAFCTLT      DS  FL1
*   _contmod_ctrltyp_set
SAFCCNO      EQU 0      none
SAFCCHC      EQU 1      hp compatible
SAFCCHP      EQU 2      hp
*
*
SAFDFFN      DS  CL62
*
SAFLINESP    DS  0XL4   _linesp
SAFCCPO      DS  H
SAFSPAC      DS  FL1
*   _line_space_set
SAFSPA1      EQU 1      space=1
SAFSPA2      EQU 2      space=2
SAFSPA3      EQU 3      space=3
SAFSPAN      EQU 4      space=no
SAFSPAЕ      EQU 5      space=ebcdic
SAFSPAИ      EQU 6      space=ibm
SAFSPAA      EQU 7      space=asa
*
SAFFIL4      DS  CL1
*
SAFDFLP      DS  H
SAFD FCC     DS  H
SAFD FHL     DS  FL1
*   _headline_set
SAFH LNO     EQU 1      no
SAFH LP      EQU 4      page number
SAFH LD      EQU 64     date
SAFH LFR     EQU 128    first record
*
SAFD FOF     DS  FL1
*   _outform_set
SAFOUTC      EQU 0      character
SAFOUTH      EQU 2      hexadecimal
*
SAFOUTPUT#   EQU *-SAFD FDF

```

..AX;

## SJAODOP - Ausgabestruktur für INFO=\*DOCUMENT\_PART

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Informationen über die unter DOCUMENT\_PART angesprochenen Optionen vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SJAODOP	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```

*   _doc_part_block_md1
SAPDPBL DS    0F
*
SAPSDPSC DS    0XL64                               Identifier of the optional
*                                                    section records
SAPSSEID DS    CL60
SAPSSEPO DS    H
SAPSSEIN DS    AL1                                sect_ind
SAPSSEGV EQU   X'80'                               ON = given, OFF = not given.
SAPSSEU1 EQU   X'7F'
SAPSSEIT DS    AL1                                id_type
SAPSSETY EQU   X'80'                               ON = character, OFF =
*                                                    hexadecimal
SAPSSEU2 EQU   X'7F'
*
*
SAPDPIP DS    0XL156                               _inpart
SAPDPDL DS    FL1
*   _select_set
SAPSEAL EQU    1
*
SAPDPU1 DS    XL3
*

```

SAPFDPFR DS	0XL76	first record
SAPFRCTY DS	AL1	type
SAPFRCBF EQU	X'80'	
SAPFRCEF EQU	X'40'	
SAPFRCBS EQU	X'20'	
SAPFRCIT EQU	X'10'	
SAPFRCU1 EQU	X'0F'	
SAPFRCSI DS	CL60	
SAPFRGIN DS	AL1	ind1
SAPFRGST EQU	X'80'	ON = character, OFF =
*		hexadecimal
SAPFRCU2 EQU	X'7F'	
SAPFRCU3 DS	XL2	
SAPFRCVA DS	F	
SAPFRCOC DS	F	
SAPFRCPO DS	H	
SAPFRCU4 DS	XL2	
*		
*		
SAPLDPLR DS	0XL76	last record
SAPLRCTY DS	AL1	type
SAPLRCBF EQU	X'80'	
SAPLRCEF EQU	X'40'	
SAPLRCBS EQU	X'20'	
SAPLRGIT EQU	X'10'	
SAPLRGU1 EQU	X'0F'	
SAPLRGSI DS	CL60	
SAPLRGIN DS	AL1	ind1
SAPLRGST EQU	X'80'	ON = character, OFF =
*		hexadecimal
SAPLRGU2 EQU	X'7F'	
SAPLRGU3 DS	XL2	
SAPLRGVA DS	F	
SAPLRGOC DS	F	
SAPLRGPO DS	H	
SAPLRGU4 DS	XL2	
*		
*		
*		
SAPRDRP DS	0XL12	record part
SAPRRPFC DS	F	
SAPRRPLC DS	F	
SAPRRPSL DS	FL1	
SAPRRPU1 DS	XL3	
*		
*		
SAPODPOP DS	0XL16	output part
SAPPOOFR DS	F	from

```
SAP0OPTO DS    F           to
SAP0OPLA DS    F           to
SAP0OPDI DS    FL1
*  _outpart_dim_set
SAPDIMN EQU    0           none
SAPDIMP EQU    1           pages
SAPDIML EQU    2           lines
*
SAP0OPSL DS    FL1
*  _outpart_select_set
SAPOSAL EQU    0           none
SAPOSRN EQU    1           yes
SAPOSLA EQU    2           no
*
SAP0OPU1 DS    XL2
*
SAPOUTPUT# EQU  *-SAPSSEID
```



## SJAOLCT - Ausgabestruktur für INFO=\*LAYOUT\_CONTROL

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Informationen über die Seitengestaltung vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SJAOLCT	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _layout_ctl_md1
SAL_LAYOUT_CTL_MDL DS    0F
SALLPGC             DS    X
SALLLMG             DS    X
SALLROT             DS    FL1
*  _rotation_set
SALROTN             EQU    0           rotation=no
SALROT0             EQU    240        rotation=0
SALROT1             EQU    241        rotation=90
SALROT2             EQU    242        rotation=180
SALROT3             EQU    243        rotation=270
SALROT4             EQU    244        rotation=0-180
SALROT5             EQU    245        rotation=90-270
SALROT6             EQU    246        rotation=180-0
SALROT7             EQU    247        rotation=270-90
SALROTY             EQU    232        rotation=by-control-codes
*
SALLTSS             DS    FL1
*  _twoside_set
SALTWYS             EQU    1           yes
SALTWTM             EQU    2           tumble
SALTWNO             EQU    4           no
*
```

```

SALLITR          DS    0XL1          in_tray
SALLIIN          DS    X
                ORG    SALLITR
SALLIIS          DS    FL1
*   _inray_set
SALITBF          EQU    0            by-format
SALITIG          EQU    1            ignore
*
                ORG    SALLITR+1
SALLIFS          DS    FL1
*   _inray_format_set
SALITMA          EQU    128          intray_format = *manual
SALITA3          EQU    129          intray_format = *a3
SALITA4          EQU    130          intray_format = *a4
SALITA5          EQU    131          intray_format = *a5
SALITB4          EQU    132          intray_format = *b4
SALITB5          EQU    133          intray_format = *b5
SALITDB          EQU    134          intray_format =
*                                     *double-letter
SALITEX          EQU    135          intray_format = *exec
SALITFO          EQU    136          intray_format = *folio
SALITIV          EQU    137          intray_format = *invoice
SALITLG          EQU    138          intray_format = *legal
SALITLT          EQU    139          intray_format = *letter
SALITMO          EQU    140          intray_format = *monarch
SALIT10          EQU    141          intray_format =
*                                     *commercial-10
SALITDL          EQU    142          intray_format = *d1
SALITC5          EQU    143          intray_format = *c5
SALIT3U          EQU    144          intray_format = *A3-uncut
SALIT4U          EQU    145          intray_format = *A4-uncut
SALITLD          EQU    146          intray_format = *ledger
*
SALLOT#          DS    0XL1          outtray
SALLOUT          DS    X
                ORG    SALLOT#
SALLOTS          DS    FL1
*   _outtray_set
SALOUTI          EQU    128          outtray = *ignore
SALOUTS          EQU    129          outtray = *sorter
*
                ORG    SALLOT#+1
SALLSRM          DS    FL1          Sorter mode
*   _srtmode_set
SALSMNO          EQU    1            srtmode = *no
SALSMGR          EQU    2            srtmode = *group
SALSMCL          EQU    3            srtmode = *collate
SALSMST          EQU    4            srtmode = *stacker

```

```
SALSM AU      EQU    5          srtmode = *automatic
*
*
SALCOVPAGE   DS     0XL40      _covpage
SALLCHP      DS     CL32
SALLCHE      DS     F
SALLCTE      DS     F
*
SALTOIN      DS     H
SALLOIN      DS     H
SAL#         EQU    *-SALLPGC
```

## SJAOPJC - Ausgabestruktur für INFO=\*PRINT\_JOB\_CONTROL

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Informationen über die Steuerung des Druckauftrags vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SJAOPJC	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _print_job_ctl_md1
SAC_PRINT_JOB_CTL_MDL DS    0X
SACPJJN                DS    CL8
SACPJPR                DS    X
SACPJCL                DS    X
SACPJMJ                DS    CL54
SACPJCH                DS    FL1
*  _checkp_set
SACCKPP                EQU    0           on pages
SACCKPR                EQU    1           on section records
*
SACPJMP                DS    FL1
*  _yes_no_set
SACYNN                EQU    0
SACYNY                EQU    1
&P.PJSD                DS    CL10        date(YYYY-MM-DD)
&P.PJST                DS    CL5         time(HH:MM)
&P.PJSS                DS    CL1         season(SUMMER/WINTER)
*
SAC#                   EQU    *-SACPJJN
```

## SJAORSD - Ausgabestruktur für INFO=\*RESOURCES\_DESCRIPTION

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Informationen über die Ressourcen vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SJAORSD	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*   _resources_descr_md1
SAR_RESOURCES_DESCR_MDL DS   0X
SARRDFN                  DS   CL6
SARRDLN                  DS   CL3
SARRDRL                  DS   CL3
*
SARRDCS                  DS   0XL53   char_set
SARRDCN                  DS   16CL3
SARRDCN#                 EQU   16
SARRDPN                  DS   CL4
SARRDPI                  DS   X
*
*
SARRDOR                  DS   0XL8   overlay_res
SARREFO                  DS   CL2
SARRFOB                  DS   CL4
*
SARRDOV                  DS   0XL2   overlay
SARRDFA                  DS   X
SARRDRE                  DS   X
*
```

```
*
SARPGDF          DS    CL8
SARF MDF         DS    CL8
SARUSRR          DS   CL44
*
SARTRANSL_TAB   DS   0XL52    _transl_tab
SARTRNA         DS    CL8
SARTRFI         DS   CL44
*
SAR#            EQU   *-SARRDFN
```

## SJAOTOP - Ausgabestruktur für INFO=\*TO\_PRINTER

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Informationen über die Ausgabegeräte vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SJAOTOP	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _to_printer_md1
SAT_TO_PRINTER_MDL DS    0X
SATPRNA             DS    CL8
SATPRTY             DS    FL1
*  _prtype_set
SATPTAN             EQU   0
SATPPTH             EQU   1
SATPTAP             EQU   2
SATPT65             EQU   3
*
SATPRVI             DS    FL1
*  _virtual_set
SATVNAL             EQU   0
SATVALL             EQU   1
SATVMUS             EQU   2
*
SATPRPN             DS    CL8
SATPRSF             DS    CL32
SATPRTG             DS    CL8
SATUN4              DS    XL2
SAT#                EQU   *-SATPRNA
```

## SPJARSO - RSO-Ausgabestruktur für SHOPJAT-Makro

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Das Makro SPJARSO beschreibt die Form, in der RSO-spezifische Werte vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SPJARSO	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
* _main_header_model
RSSA_MAIN_HEADER_MODEL DS OF
RSSADOPB DS F displacement to document part
* block
RSSADOPF DS F displacement to document
* format block
RSSAPJCB DS F displacement to print job
* control block
RSSALCOB DS F displacement to layout
* control block
RSSAREDB DS F displacement to resource
* description block
RSSATOPB DS F displacement to to printer
* block
RSSAOUTPUT# EQU *-RSSADOPB
```



## SPJADPC - Dprint-Ausgabestruktur für SHOPJAT-Makro

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der Dprint-spezifische Werte vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SPJADPC	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*   _main_header_model
DPSA_MAIN_HEADER_MODEL DS   OF
DPSADOPB                DS   F      displacement to document part
*                       block
DPSADOFF                DS   F      displacement to document
*                       format block
DPSAPJCB                DS   F      displacement to print job
*                       control block
DPSALCOB                DS   F      displacement to layout
*                       control block
DPSAREDB                DS   F      displacement to resource
*                       description block
DPSATOPB                DS   F      displacement to to printer
*                       block
DPSAOUTPUT#            EQU   *-DPSADOPB
```

## SPJASPS - SPS-Ausgabestruktur für SHOPJAT-Makro

**Benutzergruppe:** Nichtprivilegiertes Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Form, in der SPS-spezifische Werte vom Makro SHOPJAT ausgegeben werden.

### Format

Operation	Operanden
SPJASPS	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _main_header_model
SPJA_MAIN_HEADER_MODEL DS  OF
SPJADOPB                DS  F      displacement to document part
*                        block
SPJADOFF                DS  F      displacement to document
*                        format block
SPJAPJCB                DS  F      displacement to print job
*                        control block
SPJALCOB                DS  F      displacement to layout
*                        control block
SPJAREDB                DS  F      displacement to resource
*                        description block
SPJATOPB                DS  F      displacement to printer
*                        block
SPJAOUTPUT#            EQU  *-SPJADOPB
```

## 4.3 Unterstützendes Makro des SHOPJST-Makros

Das folgende Makro SPJSRSO unterstützt den Einsatz des Makros SHOJPST. Mit dem Makro SPJSRSO werden RSO-Parameter für das Makro SHOJPST gesetzt.

### SPJSRSO - RSO-Parameter für SHOPJST-Makro festlegen

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Der Makro SPJSRSO hinterlegt RSO-spezifische Werte für den Aufruf des Makros SHOPJST in einer Parameterliste.

#### Format

Operation	Operanden
SPJSRSO	XPAND=PARAM / OUTPUT  ,VARIANT= <u>001</u> / <c-string 3..3>  ,S_RSDVT= <u>*ALL</u> / array(16): *ALL / *DJET / *2030-PCL / *4011 / *4813 / *4818_PCL / *4821_PCL / *4822-PCL / *4824_PCL / *4825_PCL / *4830-PCL / *4850-PCL / *8121 / *9000 / *9000-EPFX / *9000_EPLQ / *9000_EPSQ / *9000-PCL / *9000_PRO / *9000_PS / *9001 / *9001-31 / *9002 / *9003 / *9004 / *9011 / *9012 / *9013 / *9014 / *9015 / *9021 / *9022 / *9025 / *9026-PCL / *9026_RENO / *9046 / *9097 / *9645

#### Operandenbeschreibung

##### **XPAND=PARAM / OUTPUT**

Es wird festgelegt, welche Struktur zu expandieren (erzeugen) ist.

##### **XPAND=PARAM**

Das Layout der Eingabe-Parameterliste wird expandiert.

##### **XPAND=OUTPUT**

Es werden alle Datenbeschreibungen (DSECTs) zur Beschreibung der Ausgabe-Informationenblöcke generiert.

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**S\_RSDVT=\*ALL / array(16): \*ALL / \*DJET / \*2030-PCL / \*4011 / \*4813 / \*4818\_PCL / \*4821\_PCL / \*4822-PCL / \*4824\_PCL / \*4825\_PCL / \*4830-PCL / \*4850-PCL / \*8121 / \*9000 / \*9000-EPFX / \*9000\_EPLQ / \*9000\_EPSQ / \*9000-PCL / \*9000\_PRO / \*9000\_PS / \*9001 / \*9001-31 / \*9002 / \*9003 / \*9004 / \*9011 / \*9012 / \*9013 / \*9014 / \*9015 / \*9021 / \*9022 / \*9025 / \*9026-PCL / \*9026\_RENO / \*9046 / \*9097 / \*9645**

Gerätetypen für Druckaufträge, die über RSO abgewickelt werden und über die Informationen angefordert werden. Langform des Operandennamens „select-RSO-dev-type“.

**Beschreibung der Parameterliste**

\* parameter list description

RSS\_MDL DS OF

\*

RSSSELECT DS 0XL32 select  
 RSSHDR FHDR MF=(C,RSS),EQUATES=NO standard header

\*

RSSPRDT	DS	0XL8	
RSSDTB1	DS	AL1	device type byte1
RSSDSU1	EQU	X'80'	
RSS8121	EQU	X'40'	8121
RSS8122	EQU	X'20'	8122
RSS9001	EQU	X'10'	9001
RSS9002	EQU	X'08'	9002
RSS9003	EQU	X'04'	9003
RSS9004	EQU	X'02'	9004
RSS9025	EQU	X'01'	9025
RSSDTB2	DS	AL1	device type byte2
RSS9645	EQU	X'80'	9645
RSS9031	EQU	X'40'	9013
RSS9000	EQU	X'20'	9000
RSS0131	EQU	X'10'	9001-31
RSS9011	EQU	X'08'	9011
RSS9012	EQU	X'04'	9012
RSS9022	EQU	X'02'	9022
RSS4822	EQU	X'01'	4822-PCL
RSSDTB3	DS	AL1	device type byte3
RSS4825	EQU	X'80'	4825-PCL
RSS9015	EQU	X'40'	9015
RSS9045	EQU	X'20'	9045-ANSI
RSSDSU2	EQU	X'10'	
RSS9046	EQU	X'08'	9046
RSSDSU3	EQU	X'04'	
RSS90PS	EQU	X'02'	9000-PS
RSS9PRO	EQU	X'01'	9000-PRO

RSSDTB4	DS	AL1	device type byte4
RSS9PFX	EQU	X'80'	9000-EPFX
RSS9PLQ	EQU	X'40'	9000-EPLQ
RSS9PSQ	EQU	X'20'	9000-EPSQ
RSS4818	EQU	X'10'	4818-PCL
RSSDSU4	EQU	X'0E'	
RSS9014	EQU	X'01'	9014
RSSDTB5	DS	AL1	device type byte5
RSS9021	EQU	X'80'	9021
RSS9222	EQU	X'40'	9022-200
RSS9223	EQU	X'20'	9022-300
RSS9PCL	EQU	X'10'	9000-PCL
RSSDJET	EQU	X'08'	DJET
RSS4824	EQU	X'04'	4824-PCL
RSSRENO	EQU	X'02'	9026-RENO
RSS6PCL	EQU	X'01'	9026-PCL
RSSDTB6	DS	AL1	device type byte6
RSS4812	EQU	X'80'	4812
RSS9097	EQU	X'40'	9097
RSS4011	EQU	X'20'	4011
RSS4813	EQU	X'10'	4813
RSS4821	EQU	X'08'	4821-PCL
RSS2030	EQU	X'04'	2030-PCL
RSSDSU5	EQU	X'02'	
RSS4850	EQU	x'01'	4850-PCL
RSSDTB7	DS	AL1	device type byte7
RSSD4830	EQU	x'80'	4830-PCL
RSSDSU7	EQU	X'7F'	
RSSDTB8	DS	AL1	device type byte8
RSSDSU8	EQU	X'FF'	
*			
RSSUN1	DS	XL16	
*			
RSS#	EQU	*-RSSHDR	

## 4.4 Ausgabestrukturen des SHOPJST-Makros

Die folgenden Makros beschreiben die Strukturen, mit denen der Ausgabepuffer des Makros SHOPJST nach einer Statusabfrage gefüllt wird.

„[SJSODES - Ausgabestruktur für DESTINATION](#)“ auf Seite 391

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJST nach einer Abfrage mit INFO=\*DESTINATION gefüllt wird.

„[SJSODES - Ausgabestruktur für SPOOL-FILTER](#)“ auf Seite 394

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJST nach einer Abfrage mit INFO=\*SPOOL-FILTER gefüllt wird.

„[SJSODES - Ausgabestruktur für RSO-FILTER](#)“ auf Seite 397

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJST nach einer Abfrage mit INFO=\*RSO-FILTER gefüllt wird.

„[SJSODIS - Ausgabestruktur für DISTRIBUTED](#)“ auf Seite 400

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJST nach einer Abfrage mit INFO=\*DISTRIBUTED gefüllt wird.

„[SJSODSS - Ausgabestruktur für DESTINATION und UNIX-Systeme](#)“ auf Seite 402

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST ergänzt wird, wenn eine Abfrage mit INFO=\*DESTINATION einen UNIX-System-basierten Cluster betraf.

„[SJSOORI - Ausgabestruktur für ORIGIN](#)“ auf Seite 404

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJST nach einer Abfrage mit INFO=\*ORIGIN gefüllt wird.

„[SJSOORS - Ausgabestruktur für ORIGIN und UNIX-Systeme](#)“ auf Seite 406

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST ergänzt wird, wenn eine Abfrage mit INFO=\*ORIGIN einen UNIX-System-basierten Cluster betraf.

„[SJSOTRA - Ausgabestruktur für TRAITS](#)“ auf Seite 407

Dieses Makro beschreibt die Struktur, mit der der Ausgabepuffer des Makros SHOPJST nach einer Abfrage mit INFO=\*TRAITS gefüllt wird.

„[SJSOTRS - Ausgabestruktur für TRAITS und UNIX-Systeme](#)“ auf Seite 409

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST ergänzt wird, wenn eine Abfrage mit INFO=\*TRAITS einen UNIX-System-basierten Cluster betraf.

## SJSODES - Ausgabestruktur für DESTINATION

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur des Ausgabepuffers des Makros SHOPJST, wenn der Status von Druckaufträgen mit dem Operanden INFO=\*DESTINATION abgefragt wurde. Sobald sich die Abfrage auf einen UNIX-System-basierten Cluster bezieht, wird der Ausgabepuffer um die Struktur SJSODSS ergänzt. Siehe [Abschnitt „SJSODSS - Ausgabestruktur für DESTINATION und UNIX-Systeme“ auf Seite 402](#).

### *Hinweis*

Für Druckaufträge, für die eine Scheduling-Zeit angegeben wurde, enthält ein neues Anzeigefeld die Information, dass der Auftrag verzögert wird. Für Druckaufträge, die gefiltert werden, enthält dieses Anzeigefeld die Information, dass der Auftrag konvertiert werden kann.

### Format

Operation	Operanden
SJSODES	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001** / <c-string 3..3>

Bezeichnet die Variante der generierten Parameterliste.

**Beschreibung der Parameterliste**

* _out_destination_md1			
SSD_OUT_DESTINATION_MDL	DS	0H	
SSDELEN	DS	H	block length
SSDETSN	DS	CL4	tsn
SSDESRV	DS	CL8	server name
SSDEMOD	DS	FL1	
* _out_mode_set			
SSDOMLO	EQU	1	
SSDOMRS	EQU	2	
*			
SSDESTA	DS	FL1	
* _out_job_status_set			
SSDSWTF	EQU	1	
SSDSFT	EQU	2	
SSDSWTP	EQU	3	
SSDSPRE	EQU	4	
SSDSWT	EQU	5	
SSDSTRD	EQU	6	
SSDSACT	EQU	7	
SSDSTRT	EQU	8	
SSDSKP	EQU	9	
SSSDIR	EQU	10	replay in direct mode
SSDSREL	EQU	255	
*			
SSDERPL	DS	AL1	sect_ind
SSDERPC	EQU	X'80'	ON if replay in copy mode
SSDEUNI	EQU	X'7F'	not used
SSDEDVN	DS	CL8	device name
SSDEDST	DS	CL8	destination name
SSDEECD	DS	CL8	
SSDEEMS	DS	CL8	
*			
SSDESST	DS	0XL8	
SSDDTB1	DS	AL1	device type byte1
SSDDL	EQU	X'80'	LP printer
SSDDND	EQU	X'40'	ND printer
SSDDHP	EQU	X'20'	HP printer
SSDDL	EQU	X'10'	LP48 printer
SSDDL	EQU	X'08'	LP65 printer
SSDDTAPE	EQU	X'04'	TAPE
SSDDLPEM	EQU	X'01'	LP-EMUL printer
SSDDTB2	DS	AL1	device type byte2
SSDDSU1	EQU	X'C0'	
SSDDHP90	EQU	X'20'	HP90 printer
SSDDAP50	EQU	X'10'	APA-2050 printer
SSDDAP90	EQU	X'08'	APA-2090 printer



SSDDTW90	EQU	X'04'	TWIN-2090 printer
SSDDDSU2	EQU	X'03'	
SSDDDTB3	DS	AL1	device type byte3
SSDDDSU3	EQU	X'FF'	
SSDDDTB4	DS	AL1	device type byte4
SSDDPCL	EQU	X'80'	PCL printer
SSDDDSU4	EQU	X'7F'	
SSDDDTB5	DS	AL1	device type byte5
SSDDDSU5	EQU	X'FF'	
SSDDDTB6	DS	AL1	device type byte6
SSDDDSU6	EQU	X'FF'	
SSDDDTB7	DS	AL1	device type byte7
SSDDDSU7	EQU	X'FF'	
SSDDDTB8	DS	AL1	device type byte8
SSDDDSU8	EQU	X'F0'	
SSDDVIRT	EQU	X'08'	VIRTUAL printer
SSDDDSU9	EQU	X'07'	
*			
SSDEIFG	DS	AL1	info flag
SSDEDEL	EQU	X'80'	delayed job
SSDEFLT	EQU	X'40'	filtrable job
SSD#	EQU	*-SSSLEN*	

## SJSODES - Ausgabestruktur für SPOOL-FILTER

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur des Ausgabepuffers des Makros SHOPJST, wenn der Status von Druckaufträgen mit dem Operanden INFO=\*SPOOL-FILTER abgefragt wurde.

### *Hinweis*

Für Druckaufträge, für die eine Scheduling-Zeit angegeben wurde, enthält ein neues Anzeigefeld die Information, dass der Auftrag verzögert wird. Für Druckaufträge, die gefiltert werden, enthält dieses Anzeigefeld die Information, dass der Auftrag konvertiert werden kann.

### Format

Operation	Operanden
SJSODES	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**Beschreibung der Parameterliste**

* _out_destination_md1			
SSD_OUT_DESTINATION_MDL	DS	0H	
SSDELEN	DS	H	block length
SSDETSN	DS	CL4	tsn
SSDESRV	DS	CL8	server name
SSDEMOD	DS	FL1	
* _out_mode_set			
SSDOMLO	EQU	1	
SSDOMRS	EQU	2	
*			
SSDESTA	DS	FL1	
* _out_job_status_set			
SSDSWTF	EQU	1	
SSDSFT	EQU	2	
SSDSWTP	EQU	3	
SSDSPRE	EQU	4	
SSDSWT	EQU	5	
SSDSTRD	EQU	6	
SSDSACT	EQU	7	
SSDSTRT	EQU	8	
SSDSKP	EQU	9	
SSSDIR	EQU	10	replay in direct mode
SSDSREL	EQU	255	
*			
SSDERPL	DS	AL1	sect_ind
SSDERPC	EQU	X'80'	ON if replay in copy mode
SSDEUNI	EQU	X'7F'	not used
SSDEDVN	DS	CL8	device name
SSDEDST	DS	CL8	destination name
SSDEECD	DS	CL8	
SSDEEMS	DS	CL8	
*			
SSDESST	DS	0XL8	
SSDDTB1	DS	AL1	device type byte1
SSDDL	EQU	X'80'	LP printer
SSDDND	EQU	X'40'	ND printer
SSDDHP	EQU	X'20'	HP printer
SSDDL	EQU	X'10'	LP48 printer
SSDDL	EQU	X'08'	LP65 printer
SSDDTAPE	EQU	X'04'	TAPE
SSDDLPEM	EQU	X'01'	LP-EMUL printer
SSDDTB2	DS	AL1	device type byte2
SSDDSU1	EQU	X'C0'	
SSDDHP90	EQU	X'20'	HP90 printer
SSDDAP50	EQU	X'10'	APA-2050 printer
SSDDAP90	EQU	X'08'	APA-2090 printer

SSDDTW90	EQU	X'04'	TWIN-2090 printer
SSDDDSU2	EQU	X'03'	
SSDDDTB3	DS	AL1	device type byte3
SSDDDSU3	EQU	X'FF'	
SSDDDTB4	DS	AL1	device type byte4
SSDDPCL	EQU	X'80'	PCL printer
SSDDDSU4	EQU	X'7F'	
SSDDDTB5	DS	AL1	device type byte5
SSDDDSU5	EQU	X'FF'	
SSDDDTB6	DS	AL1	device type byte6
SSDDDSU6	EQU	X'FF'	
SSDDDTB7	DS	AL1	device type byte7
SSDDDSU7	EQU	X'FF'	
SSDDDTB8	DS	AL1	device type byte8
SSDDDSU8	EQU	X'F0'	
SSDDVIRT	EQU	X'08'	VIRTUAL printer
SSDDDSU9	EQU	X'07'	
*			
SSDEIFG	DS	AL1	info flag
SSDEDEL	EQU	X'80'	delayed job
SSDEFLT	EQU	X'40'	filtrable job
SSD#	EQU	*-SSSLEN*	

## SJSODES - Ausgabestruktur für RSO-FILTER

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur des Ausgabepuffers des Makros SHOPJST, wenn der Status von Druckaufträgen mit dem Operanden INFO=\*RSO-FILTER abgefragt wurde.

### *Hinweis*

Für Druckaufträge, für die eine Scheduling-Zeit angegeben wurde, enthält ein neues Anzeigefeld die Information, dass der Auftrag verzögert wird. Für Druckaufträge, die gefiltert werden, enthält dieses Anzeigefeld die Information, dass der Auftrag konvertiert werden kann.

### Format

Operation	Operanden
SJSODES	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**Beschreibung der Parameterliste**

* _out_destination_md1			
SSD_OUT_DESTINATION_MDL	DS	0H	
SSDELEN	DS	H	block length
SSDETSN	DS	CL4	tsn
SSDESRV	DS	CL8	server name
SSDEMOD	DS	FL1	
* _out_mode_set			
SSDOMLO	EQU	1	
SSDOMRS	EQU	2	
*			
SSDESTA	DS	FL1	
* _out_job_status_set			
SSDSWTF	EQU	1	
SSDSFT	EQU	2	
SSDSWTP	EQU	3	
SSDSPRE	EQU	4	
SSDSWT	EQU	5	
SSDSTRD	EQU	6	
SSDSACT	EQU	7	
SSDSTRT	EQU	8	
SSDSKP	EQU	9	
SSSDIR	EQU	10	replay in direct mode
SSDSREL	EQU	255	
*			
SSDERPL	DS	AL1	sect_ind
SSDERPC	EQU	X'80'	ON if replay in copy mode
SSDEUNI	EQU	X'7F'	not used
SSDEDVN	DS	CL8	device name
SSDEDST	DS	CL8	destination name
SSDEECD	DS	CL8	
SSDEEMS	DS	CL8	
*			
SSDESST	DS	0XL8	
SSDDTB1	DS	AL1	device type byte1
SSDDL	EQU	X'80'	LP printer
SSDDND	EQU	X'40'	ND printer
SSDDHP	EQU	X'20'	HP printer
SSDDL	EQU	X'10'	LP48 printer
SSDDL	EQU	X'08'	LP65 printer
SSDDTAPE	EQU	X'04'	TAPE
SSDDLPEM	EQU	X'01'	LP-EMUL printer
SSDDTB2	DS	AL1	device type byte2
SSDDSU1	EQU	X'C0'	
SSDDHP90	EQU	X'20'	HP90 printer
SSDDAP50	EQU	X'10'	APA-2050 printer
SSDDAP90	EQU	X'08'	APA-2090 printer

SSDDTW90	EQU	X'04'	TWIN-2090 printer
SSDDDSU2	EQU	X'03'	
SSDDDTB3	DS	AL1	device type byte3
SSDDDSU3	EQU	X'FF'	
SSDDDTB4	DS	AL1	device type byte4
SSDDPCL	EQU	X'80'	PCL printer
SSDDDSU4	EQU	X'7F'	
SSDDDTB5	DS	AL1	device type byte5
SSDDDSU5	EQU	X'FF'	
SSDDDTB6	DS	AL1	device type byte6
SSDDDSU6	EQU	X'FF'	
SSDDDTB7	DS	AL1	device type byte7
SSDDDSU7	EQU	X'FF'	
SSDDDTB8	DS	AL1	device type byte8
SSDDDSU8	EQU	X'F0'	
SSDDVIRT	EQU	X'08'	VIRTUAL printer
SSDDDSU9	EQU	X'07'	
*			
SSDEIFG	DS	AL1	info flag
SSDEDEL	EQU	X'80'	delayed job
SSDEFLT	EQU	X'40'	filtrable job
SSD#	EQU	*-SSSLEN	

## SJSODIS - Ausgabestruktur für DISTRIBUTED

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur des Ausgabepuffers des Makros SHOPJST, wenn der Status von Druckaufträgen mit dem Operanden INFO=\*DISTRIBUTED abgefragt wurde.

### *Hinweis*

Für Druckaufträge, für die eine Scheduling-Zeit angegeben wurde, enthält ein neues Anzeigefeld die Information, dass der Auftrag verzögert wird. Für Druckaufträge, die gefiltert werden, enthält dieses Anzeigefeld die Information, dass der Auftrag konvertiert werden kann.

### Format

Operation	Operanden
SJSODIS	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.



**Beschreibung der Parameterliste**

```

*   _out_distributed_md1
SSI_OUT_DISTRIBUTED_MDL DS   0H
SSIILEN                  DS   H           block length
SSIITSN                  DS  CL4          tsn
SSIISRV                  DS  CL8          server name
SSIIPTS                  DS  CL4          tsn
SSIIJTY                  DS  FL1
*   _dist_job_type_set
SSIIJTL                  EQU   1          job-type = local
SSIIJTD                  EQU   2          job-type = dist
SSIIJTI                  EQU   3          job-type = inter
*
SSIIHST                  DS   CL8          client hostname
SSIIUID                  DS   CL8          user identification
SSIIACC                  DS   CL8          account number
SSIIECD                  DS   CL8
SSIIEMS                  DS   CL8
*
SSIIIFG                  DS   AL1          info flag
SSIIDEL                  EQU  X'80'        delayed job
SSIIFLT                  EQU  X'40'        filtrable job
SSD#                     EQU  *-SSSLEN

```

## SJSODSS - Ausgabestruktur für DESTINATION und UNIX-Systeme

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST erweitert wird, wenn sich die Abfrage mit dem Operanden INFO=\*DESTINATION auf einen UNIX-System-basierten Cluster bezieht.

### Format

Operation	Operanden
SJSODSS	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**Beschreibung der Parameterliste**

```

*   _out_destination_sinix_md1
SID_OUT_DESTINATION_SINIX_MDL DS   0H
SIDELEN                        DS   H   block length
SIDESSN                        DS   CL8  server name
SIDESST                         DS   FL1
*   _out_job_status_set
SIDSWTF                        EQU   1
SIDSFT                         EQU   2
SIDSWTP                        EQU   3
SIDSPRE                        EQU   4
SIDSWT                         EQU   5
SIDSTRD                        EQU   6
SIDSACT                        EQU   7
SIDSTRT                        EQU   8
SIDSKP                         EQU   9
SIDSDIR                        EQU  10   replay in direct mode
SIDREL                         EQU  255
*
SIDESDN                        DS   CL8  device name
SIDESID                        DS   CL255
*                               identification
SIDESUN                        DS   XL2
SID#                           EQU   *-SIDELEN

```

## SJSOORI - Ausgabestruktur für ORIGIN

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur des Ausgabepuffers des Makros SHOPJST, wenn der Status von Druckaufträgen mit dem Operanden INFO=\*ORIGIN abgefragt wurde. Sobald sich die Abfrage auf einen UNIX-System-basierten Cluster bezieht, wird der Ausgabepuffer um die Struktur SJSOORS ergänzt. Siehe [Abschnitt „SJSOORS - Ausgabestruktur für ORIGIN und UNIX-Systeme“](#) auf Seite 406.

### Format

Operation	Operanden
SJSOORI	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _out_origin_md1
SSO_OUT_ORIGIN_MDL DS   OH
SSOOLEN             DS   H           block length
SSO0TSN             DS   CL4         tsn
SSO0SRV             DS   CL8         server name
SSO0SPN             DS   CL8         spoolout name
SSO0RTS             DS   CL4         tsn of the requestor
SSO0HST             DS   CL8         client hostname
SSO0UID             DS   CL8         user identification
SSO0ACC             DS   CL8         account number
SSO0FCO             DS   X           number of copies asked
SSO0PCO             DS   X           number of page copies asked
SSO0FTY             DS   FL1         file type
*  _out_file_type_set
SSODMS              EQU   1
SSOEAMF             EQU   2
SSOOMF              EQU   3
SSOPLAM             EQU   4
SSOLST              EQU   5
SSOOPT              EQU   6
```

SS00UT	EQU	7	
SS0TMP	EQU	8	
SS0UFS	EQU	9	
*			
SS00FCB	DS	FL1	FCB type
* _out_file_type_set			
SS0SAM	EQU	0	
SS0TAPE	EQU	1	replay file
SS0EAM	EQU	2	
SS0ISAM	EQU	64	
SS0BTAM	EQU	128	
SS0PAM	EQU	192	
*			
SS00FSZ	DS	H	file size in PAM-pages
SS0#	EQU	*-SS00LEN	

## SJSOORS - Ausgabestruktur für ORIGIN und UNIX-Systeme

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST erweitert wird, wenn sich die Abfrage mit dem Operanden INFO=\*ORIGIN auf einen UNIX-System-basierten Cluster bezieht.

### Format

Operation	Operanden
SJSOORS	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _out_origin_sinix_md1
SIO_OUT_ORIGIN_SINIX_MDL DS  OH
SIO0LEN                   DS  H           block length
SIO0SRV                   DS  CL8        server name
SIO0SSN                   DS  CL8        spoolout name
SIO0SHN                   DS  CL8        client hostname
SIO0SUI                   DS  CL8        user identification
SIO0SFC                   DS  X          number of copies asked
SIO0SPC                   DS  X          number of page copies asked
SIO0SID                   DS  CL255     identification
SIOUNI                    DS  XL1
SIO#                      EQU  *--SIO0LEN
```

## SJSOTRA - Ausgabestruktur für TRAITS

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur des Ausgabepuffer des Makros SHOPJST, wenn der Status von Druckaufträgen mit dem Operanden INFO=\*TRAITS abgefragt wurde. Sobald sich die Abfrage auf einen UNIX-System-basierten Cluster bezieht, wird der Ausgabepuffer um die Struktur SJSOTRS ergänzt. Siehe [Abschnitt „SJSOTRS - Ausgabestruktur für TRAITS und UNIX-Systeme“ auf Seite 409](#).

### Format

Operation	Operanden
SJSOTRA	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _out_traits_md1
SST_OUT_TRAITS_MDL DS    0H
SSTLEN             DS    H           block length
SSTTSN            DS    CL4         tsn
SSTTSRV           DS    CL8         server name
SSTTPRI           DS    X           priority
SSTTFRM           DS    CL8         form name
SSTTCLA           DS    X           job class
SSTTCSN           DS    X           character set number
SSTTUN1           DS    XL1
SSTTFOB           DS    H           fob size
SSTTEFO           DS    CL2         form overlay
SSTTROT           DS    FL1         rotation
*  _out_rotation_set
SSTOR0            EQU    0           rotation = 0
SSTOR90           EQU    1           rotation = 90
SSTOR18           EQU    2           rotation = 180
SSTOR27           EQU    3           rotation = 270
SSTOR01           EQU    4           rotation = 0
SSTOR92           EQU    5           rotation = 90
```

SSTOR10	EQU	6	rotation = 180
SSTOR29	EQU	7	rotation = 270
SSTORY	EQU	8	rotation = *yes
SSTORNO	EQU	9	rotation = *no
*			
SSTTCTL	DS	FL1	control output
* _out_control_set			
SSTOCAP	EQU	2	control = apa
SSTOCTR	EQU	4	control=transparent
SSTOCLI	EQU	8	control = line
SSTPCLG	EQU	32	control = logical
SSTOCPH	EQU	64	control = physical
SSTOCNO	EQU	128	control = no
*			
SSTTRES	DS	AL1	resource_location
SSTTRSV	EQU	X'80'	ON : resource on server - OFF
*			: resource on client
SSTTUN2	EQU	X'7F'	not used
SSTTUN3	DS	XL1	
SST#	EQU	*-SSTLEN	



## SJSOTRS - Ausgabestruktur für TRAITS und UNIX-Systeme

**Benutzergruppe:** Nichtprivilegierter Benutzer

**Programmiersprache:** Assembler

**Makrotyp:** S

Dieses Makro beschreibt die Struktur, um die der Ausgabepuffer des Makros SHOPJST erweitert wird, wenn sich die Abfrage mit dem Operanden INFO=\*TRAITS auf einen UNIX-System-basierten Cluster bezieht.

### Format

Operation	Operanden
SJSOTRS	VARIANT= <u>001</u> / <c-string 3..3>

### Operandenbeschreibung

**VARIANT=001 / <c-string 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

### Beschreibung der Parameterliste

```
*  _out_traits_sinix_md1
SIT_OUT_TRAITS_SINIX_MDL DS   OH
SITTSLE                   DS   H           block length
SITTSSN                   DS   CL8        server name
SITTSPR                   DS   X           number of copies asked
SITTSFN                   DS   CL8        form name
SITTSRT                   DS   FL1        rotation
*  _out_rotation_set
SITOR0                    EQU   0         rotation = 0
SITOR90                   EQU   1         rotation = 90
SITOR18                   EQU   2         rotation = 180
SITOR27                   EQU   3         rotation = 270
SITOR01                   EQU   4         rotation = 0
SITOR92                   EQU   5         rotation = 90
SITOR10                   EQU   6         rotation = 180
SITOR29                   EQU   7         rotation = 270
SITORY                    EQU   8         rotation = *yes
SITORNO                   EQU   9         rotation = *no
*
SITTSID                   DS   CL255      identification
SITTUN3                   DS   XL1
SIT#                      EQU   *-SITTSLE
```



---

## 5 Makros für die Verwaltung von Druckern

Für die Verwaltung von Druckern wird eine Programmschnittstelle angeboten, die fünf Makros umfasst. Diese Makros bieten dieselbe Funktionalität wie die entsprechenden Kommandos:

- SASDSPO(x) entspricht dem Kommando SHOW-ACTIVE-SPOOL-DEVICES.  
Siehe „[SASDSPO - Informationen über aktive SPOOL-Geräte anfordern](#)“ auf Seite 412.
- STPSPPR(x) entspricht dem Kommando STOP-PRINTER-OUTPUT.  
Siehe „[STPSPPR - Druckerzuweisung aufheben](#)“ auf Seite 452.
- STRSPPR(x) entspricht dem Kommando START-PRINTER-OUTPUT.  
Siehe „[STRSPPR - Druckern Druckaufträge zuweisen](#)“ auf Seite 457.
- STRRSO(x) entspricht den RSO spezifische Operanden, die mit dem STRSPPR-Makro benutzt werden. Siehe „[STRRSO - Drucker mit RSO-spezifischen Parametern zuweisen](#)“ auf Seite 475.
- SUPSPPR(x) entspricht dem Kommando MODIFY-PRINTER-OUTPUT-STATUS.  
Siehe „[SUPSPPR - Verwaltungsparameter für Drucker ändern](#)“ auf Seite 478.

(x) ist die Ergänzung gemäß der benutzten Programmiersprache (ein Leerzeichen steht für Assembler, „C“ für CPP und „H“ für C).

## SASDSPO - Informationen über aktive SPOOL-Geräte anfordern

**Benutzergruppe:** Nichtprivilegierter Benutzer, RSO-Geräteverwalter, SPOOL-Verwalter, Cluster-Verwalter

**Programmiersprache:** Assembler, CPP, C

**Makrotyp:** M, S

Dieses Makro gibt Informationen über die bezeichneten Drucker aus (z. B. für Drucker, für die ein START-PRINTER-OUTPUT-Kommando abgegeben wurde). Der Drucker, über den Informationen gewünscht werden, kann innerhalb oder außerhalb der Cluster-Konfiguration angesiedelt sein, zu der der Benutzer gehört.

### Format

Operation	Operanden
SASDSPO	ENTRY = <u>YES</u> / NO  XPAND = PARAM / OUTPUT  VARIANT = <c-string-without-quotes 3..3> / <u>001</u>  CALLER = <u>*USER</u> / *SYSTEM  INFO = <var: enum-of _information_set:1> / <u>*SUMMARY</u> / *ALL  DPCPAR = <var: pointer> / <u>*NONE</u>  RSOPAR = <var: pointer> / <u>*NONE</u>  SPSPAR = <var: pointer> / <u>*NONE</u>  DSEMPAR = <var: pointer> / <u>*NONE</u>  BUFFER = <var: pointer> / <u>*NONE</u>  BUFLen = <var: int:4> / <integer 1..8388607> / <u>*NONE</u>

(Teil 1 von 4)

Operation	Operanden
SASDSPO	<p>SCHEDST = &lt;var: enum-of _sched_state_set:1&gt; / <u>*NEXT</u> / *CURRENT</p> <p>ROT = <u>*ANY</u> / *YES / *NO / *MANUAL</p> <p>TWO_UP = <u>*ANY</u> / *YES / *NO / *MODE1 / *MODE2</p> <p>EXITR = <u>*ANY</u> / *YES / *NO</p> <p>STATUS = <u>*ALL</u> / *ATTACHED / *IDLE / *TRANSFER / *RUN / *MESSAGE / *STOPPED / *WORKED / *DETACHED / *UNKNOWN</p> <p>ERR_TYP = &lt;var: enum-of _flag_set:1&gt; / *ALL / *LIST / *WILD / <u>*ANY</u> / *NONE / *CLIST</p> <p>ERR_MSG = array(1): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; / <u>*ALL</u></p> <p>EMSG_W = &lt;var: char:24&gt; / &lt;c-string 1..24: c-string 1-24&gt; / <u>*ALL</u></p> <p>DVC_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD / <u>*ALL</u> / *CLIST</p> <p>DVCNAME = array(8): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; / <u>*ALL</u></p> <p>DVCN_W = &lt;var: char:24&gt; / &lt;c-string 1..24: c-string 1-24&gt; / <u>*ALL</u></p> <p>DES_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / <u>*LOCAL</u> / *REMOTE / *PUBLIC / *ALL</p> <p>DESNAME = array(16): &lt;var: char:8&gt; 7 &lt;c-string 1..8: c-string 1-8&gt; / <u>*ALL</u></p> <p>FRM_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *STD 7 *EXCEPT / <u>*ALL</u></p>

(Teil 2 von 4)

Operation	Operanden
SASDSPO	<p>FRMNAME = array(16): &lt;var: char:6&gt; / &lt;c-string 1..6: c-string 1-6&gt; / <u>*ALL</u></p> <p>USR_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *EXCEPT / <u>*ALL</u> / *CLIST</p> <p>USRNAME = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; / <u>*ALL</u></p> <p>CLS_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *EXCEPT / <u>*ALL</u></p> <p>CLASS = array(16): &lt;var: int:2&gt; / &lt;integer 1..255&gt; / <u>*ALL</u></p> <p>SPN_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *EXCEPT / <u>*ALL</u> / *CLIST</p> <p>SPNAME = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; / <u>*ALL</u></p> <p>ACC_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *EXCEPT / <u>*ALL</u></p> <p>ACCOUNT = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; / <u>*ALL</u></p> <p>OVL_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *NONE / *ONLY / *EXCEPT / <u>*ALL</u></p> <p>OVERLAY = array(16): &lt;var: char:2&gt; / &lt;c-string 2..2: c-string 2-2&gt; / <u>*ALL</u></p> <p>FOB = (<i>type, low, high</i>)  <i>type</i> : <u>*ANY</u> / *ONLY / *NO / *RANGE  <i>low</i> : &lt;var: int:2&gt; / &lt;integer 0..32767&gt; / <u>*MIN</u>  <i>high</i> : &lt;var: int:2&gt; / &lt;integer 0..32767&gt; / <u>*STD</u></p>

(Teil 3 von 4)

Operation	Operanden
SASDSPO	<p>PRIO = (type, low, high)                      type : <u>*ALL</u> / *RANGE                      low : &lt;var: int:1&gt; / &lt;integer 30..255&gt; / <u>*MIN</u>                      high: &lt;var: int:1&gt; / &lt;integer 30..255&gt; / <u>*MAX</u></p> <p>CSN = (type, low, high)                      type : <u>*ALL</u> / *ONE / *RANGE                      low : &lt;var: int:2&gt; / &lt;integer 1..32767&gt; / <u>*MIN</u>                      high : &lt;var: int:2&gt; / &lt;integer 1..32767&gt; / <u>*MAX</u></p> <p>SRV_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST / *WILD / <u>*ALL</u> /                      *HOME</p> <p>SERVER = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; /  <u>*ALL</u></p> <p>SRW_WLD = &lt;var: char:24&gt; / &lt;c-string 1..24: c-string 1-24&gt; / <u>*ALL</u></p> <p>HST_TYP = &lt;var: enum-of _flag_set:1&gt; / *LIST /                      *LOCAL_CLUSTER / <u>*ALL</u> / *HOME / *EXCEPT / *CLIST</p> <p>HSTNAME = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt; /  <u>*ALL</u></p> <p>CLUSTER = &lt;var: char:8&gt; / &lt;c-string 1..8: name_1..8&gt; /  <u>*LOCAL_CLUSTER</u></p>

(Teil 4 von 4)

### Operandenbeschreibung

„SPOOL-Gerät“ bedeutet im folgenden Text entweder lokales Gerät (verteiltes oder nicht-verteiltes) oder RSO-Gerät.

#### ENTRY=YES/NO

Angabe zu korrespondierenden Einträgen.

#### ENTRY=YES

Korrespondierende Einträge werden generiert.

#### ENTRY=NO

Korrespondierende Einträge werden nicht generiert.

**XPAND=PARAM / OUTPUT**

Es wird festgelegt, welche Struktur zu expandieren (erzeugen) ist.

**XPAND=PARAM**

Das Layout der Eingabe-Parameterliste wird expandiert.

**XPAND=OUTPUT**

Es werden alle Datenbeschreibungen (DSECTs) zur Beschreibung der Ausgabe-Informationsblöcke generiert.

**VARIANT=001 / <c-string\_without\_quotes 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**CALLER=\*USER / \*SYSTEM**

Aufrufer des Makros.

**CALLER=\*USER**

Vorgabewert für diesen Operanden. Ein Benutzer ruft das Makro auf.

**CALLER=\*SYSTEM**

Vorgabewert für diesen Operanden. Das Makro wird vom System aufgerufen

**INFO=<var: enum-of \_information\_set:1> / \*SUMMARY / \*ALL**

Mit diesem Operanden kann ausgewählt werden, welche Informationen ausgegeben werden sollen.

**INFO=<var: enum-of \_information\_set:1>**

Die Informationen, die ausgegeben werden sollen, werden nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integer-Wert gibt dabei den Typ an. Es herrscht folgende Beziehung zwischen Wert und Funktionstyp:

0 \* SUMMARY

1 \* ALL

**INFO=\*SUMMARY**

Es wird für jedes Element nur eine Zeile angegeben.

**INFO=\*ALL**

Für jedes Element werden alle Zeilen ausgegeben.

**DPCPAR=\*NONE/<var: pointer>**

Es kann die Adresse einer Parameterliste angegeben werden, in der DPRINT-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**DPCPAR=\*NONE**

Vorgabewert für diesen Operanden. Die Werte sollen nicht über eine Parameterliste versorgt werden.



**DPCPAR=<var: pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die DPRINT-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die DPRINT-Parameterliste hinterlegt ist (A(feld)).

**RSOPAR=\*NONE / <var: pointer>**

Es kann die Adresse einer Parameterliste angegeben werden, in der RSO-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**RSOPAR=\*NONE**

Vorgabewert für diesen Operanden. Die Werte sollen nicht über eine Parameterliste versorgt werden.

**RSOPAR=<var: pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die RSO-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die RSO-Parameterliste hinterlegt ist (A(feld)).

**SPSPAR=\*NONE/<var: pointer>**

Es kann die Adresse einer Parameterliste angegeben werden, in der SPS-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**SPSPAR=\*NONE**

Vorgabewert für diesen Operanden. Die Werte sollen nicht über eine Parameterliste versorgt werden.

**SPSPAR=<var: pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die SPS-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die SPS-Parameterliste hinterlegt ist (A(feld)).

**DSEMPAR=\*NONE/<var: pointer>**

Es kann die Adresse einer Parameterliste angegeben werden, in der DSEM-spezifische Werte für den Aufruf des Makros hinterlegt sind.

**DSEMPAR=\*NONE**

Vorgabewert für diesen Operanden. Die Werte sollen nicht über eine Parameterliste versorgt werden.

**DSEMPAR=<var: pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die DSEM-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die DSEM-Parameterliste hinterlegt ist (A(feld)).

**BUFFER=<var: pointer>**

Es wird ein Zeiger auf den Ausgabepuffer angegeben. Es muss zusätzlich auch die Größe des Puffers angegeben werden (siehe BUFLen). Wenn diese zu kurz ist, wird ein Fehler zurückgegeben.

**BUFFER=\*NONE**

Vorgabewert für diesen Operanden. Es wird kein Ausgabepuffer angegeben. Bei BUFLLEN muss darum auch \*NONE angegeben sein. In diesem Fall fragt Spool & Print selbst den geeigneten Puffer im Klasse-6-Speicher an. Die Adresse dieses Puffers wird zur Eingabeparameter-Liste zurückgegeben, genauso wie die Länge des Puffers. In diesem Fall muss der Puffer vom Aufrufer freigegeben werden.

**BUFLLEN=<var: int: 4> / <integer 1..8388607>**

Größe des Ausgabepuffers.

**BUFLLEN=<var: int: 4>**

Die Größe des Ausgabepuffers wird nicht direkt über den entsprechenden Operandenwert, sondern indirekt über ein Feld angegeben.

**BUFLLEN=<integer 1..8388607>**

Die Größe des Ausgabepuffers wird direkt angegeben.

**BUFLLEN=\*NONE**

Vorgabewert für diesen Operanden. Es wird keine Größe des Ausgabepuffers angegeben.

**SCHEDST=\*NEXT/\*CURRENT/<var: enum-of \_sched\_state\_set:1>**

Definiert den Typ des Eintrags für den Scheduling-Status.

**SCHEDST=\*NEXT**

Vorgabewert für diesen Operanden. Es wird die nächste Scheduling-Operation ausgewählt.

**SCHEDST=\*CURRENT**

Es wird die gegenwärtige Scheduling-Operation ausgewählt.

**SCHEDST=<var: enum-of \_sched\_state\_set:1>**

Der Scheduling-Status wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Status an. Es herrscht folgende Beziehung zwischen Wert und Status:

- 1 \*CURRENT
- 2 \*NEXT

**ROT=\*ANY/\*YES/\*NO/\*MANUAL**

Es werden Informationen über ein aktives SPOOL-Gerät ausgegeben, gemäß dem Eintrag im ROTATION-Operanden des START-PRINTER-OUTPUT-, des START-TAPE-OUTPUT oder des START-TAPE-REPLAY-Kommandos.

**ROT=\*ANY**

Vorgabewert für diesen Operanden. Es werden Informationen über aktive SPOOL-Geräte ausgegeben, egal ob an ihnen Druckaufträge das Seitenrotationmodul benutzen oder nicht.

**ROT=\*YES**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, bei denen Druckaufträge das Seitenrotationmodul benutzen.

**ROT=\*NO**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, bei denen Druckaufträge das Seitenrotationmodul nicht benutzen.

**ROT=\*MANUAL**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, bei denen Druckaufträge auf das Seitenrotationmodul manuell zugreifen.

**TWO\_UP=\*ANY/\*YES/\*NO/\*MODE1/\*MODE2**

Es werden Informationen über Drucker ausgegeben, für die der korrespondierende Wert TWO-UP-PROCESSING im START-PRINTER-OUTPUT-Kommando spezifiziert ist.

Weitere Angaben siehe im Handbuch „[SPOOL \(BS2000/OSD\)](#)“.

**EXITR=\*ANY/\*YES/\*NO**

Es werden Informationen über ein aktives SPOOL-Gerät ausgegeben, gemäß dem Eintrag im EXIT-ROUTINE -Operanden des START-PRINTER-OUTPUT-Kommandos.

**EXITR=\*ANY**

Vorgabewert für diesen Operanden. Es werden Informationen über aktive SPOOL-Geräte ausgegeben, egal ob sie die Exitroutine benutzen oder nicht.

**EXITR=\*YES**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, die die Exitroutine benutzen.

**EXITR=\*NO**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, die die Exitroutine nicht benutzen.

**STATUS=\*ALL/\*ATTACHED/\*IDLE/\*TRANSFER/\*RUN/\*MESSAGE/\*STOPPED/  
\*WORKED/\*DETACHED/\*UNKNOWN**

Es werden Informationen über ein aktives SPOOL-Gerät ausgegeben, bei denen der Status dem angegebenen Wert entspricht.

**ERR\_TYP = <var: enum-of \_flag\_set:1> / \*ALL / \*LIST / \*WILD / \*ANY / \*NONE / \*CLIST**  
Legt den Typ des Eintrags für die Auswahl des Fehlermeldungsoperanden fest.

**ERR\_TYP=<var: enum-of \_flag\_set:1>**

Der Typ des Eintrags für ERR\_MSG and EMSG\_W wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**ERR\_TYP=\*ALL**

Alle Fehlermeldungen sind gültig.

**ERR\_TYP=\*LIST**

Gibt eine Liste von Fehlermeldungen an.

**ERR\_TYP=\*WILD**

Die Fehlermeldungen werden mit Platzhaltern teilqualifiziert festgelegt.

**ERR\_TYP=\*ANY**

Vorgabewert für diesen Operanden. Alle aktiven SPOOL-Geräte werden ausgewählt, egal ob sie eine Fehlermeldung haben oder nicht.

**ERR\_TYP=\*NONE**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, für die es keine Fehlermeldung gibt.

**ERR\_TYP=\*CLIST**

Alle Elemente der Liste der Fehlermeldungen müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**ERR\_MSG=\*ALL/array(1): <var: char:8>/<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, für die es eine Fehlermeldung gibt.

**ERR\_MSG=\*ALL**

Vorgabewert für diesen Operanden. Es werden Informationen über aktive SPOOL-Geräte für alle Fehlermeldungen ausgegeben.

**ERR\_MSG=array(1): <var: char:8>**

Es werden Informationen über aktive SPOOL-Geräte für die angegebene Liste von Fehlermeldungen angezeigt.

**ERR\_MSG=<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive SPOOL-Geräte für die angegebene Fehlermeldung angezeigt.

**EMSG\_W=\*ALL/<var: char:24>/<c-string 1..24: c-string 1-24>**

Es werden Informationen über aktive SPOOL-Geräte für die angegebenen Fehlermeldungen angezeigt. Die Angabe von Platzhaltern ist erlaubt.

**EMSG\_W=\*ALL**

Vorgabewert für diesen Operanden. Es werden Informationen über aktive SPOOL-Geräte für alle Fehlermeldungen ausgegeben.

**EMSG\_W=<var: char:24>/<c-string 1..24: c-string 1-24>**

Die Fehlermeldungen werden mit Platzhaltern teilqualifiziert festgelegt.

**DVC\_TYP=<var: enum-of \_flag\_set:1> / \*LIST / \*WILD / \*ALL / \*CLIST**

Legt den Typ des Eintrags für den Gerätenamen fest.

**DVC\_TYP=<var: enum-of \_flag\_set:1>**

Der Typ des Eintrags für DVCN\_NAME and DVCN\_W wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**DVC\_TYP=\*LIST**

Eine Liste von Gerätenamen wird spezifiziert.

**DVC\_TYP=\*WILD**

Die Gerätenamen werden teilqualifiziert mit Platzhaltern angegeben.

**DVC\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Gerätenamen sind gültig.

**DVC\_TYP=\*CLIST**

Alle Elemente der Liste der Gerätetypen müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**DVCNAME=\*ALL/array(8): <var: char:8>/<c-string 1..8: c-string 1-8>**

Namen der aktiven SPOOL-Geräte, für die Informationen angegeben werden.

**DVCNAME=\*ALL**

Vorgabewert für diesen Operanden. Es werden (Kurz-)Informationen über alle Geräte, zu denen der Benutzer Zugriff hat, angefordert.

**DVCNAME=array(8): <var: char:8>**

Es werden Informationen über alle Geräte der spezifizierten Liste angefordert.

**DVCNAME=<c-string 1..8: c-string 1-8>**

Es werden Informationen über das angegebene Gerät angefordert.

**DVCN\_W=\*ALL/<var: char:24>/<c-string 1..24: c-string 1-24>**

Namen der aktiven SPOOL-Geräte, für die Informationen angefordert werden. Die Angabe von Platzhaltern ist erlaubt.

**DVCN\_W=\*ALL**

Vorgabewert für diesen Operanden. Es werden Informationen über alle Geräte angefordert.

**DVCN\_W=<var: char:24>/<c-string 1..24: c-string 1-24>**

Die Gerätenamen werden teilqualifiziert mit Platzhaltern angegeben.

**DES\_TYP=\*LOCAL/<var: enum-of \_flag\_set:1>/\*LIST/\*REMOTE/\*PUBLIC/\*ALL**

Es werden Informationen über aktive SPOOL-Geräte ausgegeben, gemäß dem Eintrag im DESTINATION-Operanden des START-PRINTER-OUTPUT-Kommandos oder des spezifizierten aktiven Drucker.

**DES\_TYP=\*LOCAL**

Vorgabewert für diesen Operanden. Es werden lokale Geräte ausgewählt (verteilt oder nicht-verteilt).

**DES\_TYP=<var: enum-of \_flag\_set:1>**

DES\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**DES\_TYP=\*LIST**

Eine Liste der zugeordneten Geräte wird spezifiziert.

**DES\_TYP=\*REMOTE**

Es werden aktive RSO-Geräte ausgewählt. Dieser Operand darf nur in Verbindung mit CLUSTER-NAME=\*LOCAL-CLUSTER und SERVER-NAME=\*ALL oder \*HOME angegeben werden.

**DES\_TYP=\*PUBLIC**

Es werden aktive „public“ RSO-Geräte ausgewählt. Dieser Operand darf nur in Verbindung mit CLUSTER-NAME=\*LOCAL-CLUSTER und SERVER-NAME=\*ALL or \*HOME angegeben werden.

**DES\_TYP=\*ALL**

Es werden alle Geräte ausgewählt, egal ob lokal oder RSO. Wenn RSO nicht verteilt ist, werden nur aktive RSO-Geräte der lokalen SPOOL-Parameterdatei ausgewählt.

**DESNAME=\*ALL/array (16): <var: char: 8>/<c-string 1..8: c-string 1-8>**

Legt den Namen der Geräte-Pools fest, über den Informationen angefordert werden.

**DESNAME=\*ALL**

Vorgabewert für diesen Operanden. Es werden Informationen über alle Geräte-Pools angefordert.

**DESNAME=array(16): <var: char:8>**

Liste von Geräte-Pools, über die Informationen angefordert werden.

**DESNAME=<c-string 1..8: c-string 1-8>**

Name des Geräte-Pools, über den Informationen angefordert werden.

**FRM\_TYP=\*ALL/<var: enum-of \_flag\_set:1>/\*LIST/\*STD/\*EXCEPT**

Legt den Typ der Angabe für Formulare fest.

**FRM\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Formulare sind gültig

**FRM\_TYP=<var: enum-of \_flag\_set:1>**

FRM\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**FRM\_TYP=\*LIST**

Eine Liste von Formulare wird ausgewählt.

**FRM\_TYP=\*STD**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten Formulare zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**FRM\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte angefordert, für die die spezifizierten Formulare ausgeschlossen sind.

**FRMNAME=\*ALL/array(16): <var: char:6>/<c-string 1..6: c-string 1-6>**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten Formulare zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**FRMNAME=\*ALL**

Vorgabewert für diesen Operanden. Alle Formularnamen werden ausgewählt.

**FRMNAME=array(16): <var: char:6>**

Eine Liste von Formularnamen, die den Geräten zugeordnet oder für sie ausgeschlossen sind, über die Informationen ausgegeben werden.

**FRMNAME=<c-string 1..6: c-string 1-6>**

Formularname (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**USR\_TYP=<var: enum-of \_flag\_set:1> / \*LIST / \*EXCEPT / \*ALL / \*CLIST**

Legt den Typ des Eintrags für die Benutzerkennungen der Druckaufträge fest.

**USR\_TYP=<var: enum-of \_flag\_set:1>**

USR\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**USR\_TYP=\*LIST**

Liste der Benutzerkennungen der Druckaufträge, über die Informationen ausgegeben werden sollen.

**USR\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte angefordert, für die die spezifizierten Benutzerkennungen ausgeschlossen sind.

**USR\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Benutzerkennungen werden ausgewählt.

**USR\_TYP=\*CLIST**

Alle Elemente der Liste der Benutzerkennungen müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**USRNAME=\*ALL/array(16): <var: char:8>/<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten Benutzerkennungen zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**USRNAME=\*ALL**

Vorgabewert für diesen Operanden. Alle Druckaufträge werden ausgewählt.

**USRNAME=array(16): <var: char:8>**

Liste der Benutzerkennungen für die aktiven Geräte, über die Informationen ausgegeben werden sollen.

**USERNAME=<c-string 1..8: c-string 1-8>**

Benutzerkennung (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**CLS\_TYP=\*ALL/<var: enum-of \_flag\_set:1>/\*LIST/\*EXCEPT**

Legt den Typ der Einträge für SPOOLOUT-Klassen fest.

**CLS\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle SPOOLOUT-Klassen werden ausgewählt.

**CLS\_TYP=<var: enum-of \_flag\_set:1>**

CLS\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**CLS\_TYP=\*LIST**

Liste der SPOOLOUT-Klassen für die aktiven Geräte, über die Informationen ausgegeben werden sollen.

**CLS\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte angefordert, für die die spezifizierten SPOOLOUT-Klassen ausgeschlossen sind

**CLASS=\*ALL/array (16): <var: int: 2>/<integer 1..255>**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten SPOOLOUT-Klassen zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**CLASS=\*ALL**

Vorgabewert für diesen Operanden. Alle SPOOLOUT-Klassen werden ausgewählt.

**CLASS=array (16): <var: int: 2>**

Liste der SPOOLOUT-Klassen (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**CLASS=<integer 1..255>**

SPOOLOUT-Klasse (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**SPN\_TYP=<var: enum-of \_flag\_set:1> / \*LIST / \*EXCEPT / \*ALL / \*CLIST**

Legt den Typ des Eintrags für die Job-Namen fest.

**SPN\_TYP=<var: enum-of \_flag\_set:1>**

SPN\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**SPN\_TYP=\*LIST**

Liste der Job-Namen für die aktiven Geräte, über die informiert werden soll.



**SPN\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte angefordert, für die die spezifizierten Job-Namen ausgeschlossen sind.

**SPN\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Job-Namen werden ausgewählt.

**SPN\_TYP=\*CLIST**

Alle Elemente der Liste der Auftragsnamen müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**SPNAME=\*ALL/array(16): <var: char:8>/<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten Job-Namen zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**SPNAME=\*ALL**

Vorgabewert für diesen Operanden. Alle Job-Namen werden ausgewählt.

**SPNAME=array(16): <var: char:8>**

Liste der Job-Namen (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**SPNAME=<c-string 1..8: c-string 1-8>**

Job-Name (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**ACC\_TYP=\*ALL/<var: enum-of \_flag\_set:1>\*/LIST\*/EXCEPT**

Legt den Typ des Eintrags für die Abrechnungsnummer fest.

**ACC\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Abrechnungsnummern werden ausgewählt.

**ACC\_TYP=<var: enum-of \_flag\_set:1>**

ACC\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**ACC\_TYP=\*LIST**

Liste der Abrechnungsnummern für die aktiven Geräte, über die informiert werden soll.

**ACC\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte angefordert, für die die spezifizierten Abrechnungsnummern ausgeschlossen sind.

**ACCOUNT=\*ALL/array (16): <var: char: 8>/<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten Abrechnungsnummern zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**ACCOUNT=\*ALL**

Vorgabewert für diesen Operanden. Alle Abrechnungsnummern werden ausgewählt.

**ACCOUNT=array(16): <var: char:8>**

Liste der Abrechnungsnummern (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**ACCOUNT=<c-string 1..8: c-string 1-8>**

Abrechnungsnummer (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**OVL\_TYP=\*ALL/<var: enum-of \_flag\_set:1>/\*LIST/\*NONE/\*ONLY/\*EXCEPT**

Legt den Typ des Eintrags für die Filmdias fest.

**OVL\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Filmdias werden ausgewählt.

**OVL\_TYP=<var: enum-of \_flag\_set:1>**

OVL\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**OVL\_TYP=\*LIST**

Liste der Filmdias für die aktiven Geräte, über die informiert werden soll.

**OVL\_TYP=\*NONE**

Es werden Informationen über aktive Geräte angefordert, die keine Filmdias unterstützen.

**OVL\_TYP=\*ONLY**

Es werden Informationen über aktive Geräte angefordert, die Filmdias unterstützen.

**OVL\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte angefordert, für die die spezifizierten Filmdias ausgeschlossen sind.

**OVERLAY=\*ALL/array (16): <var: char: 2>/<c-string 1..2: c-string 1-2>**

Es werden Informationen über aktive Geräte angefordert, denen die spezifizierten Filmdias zugeordnet sind oder für die sie ausgeschlossen sind (siehe START-PRINTER-OUTPUT).

**OVERLAY=\*ALL**

Vorgabewert für diesen Operanden. Alle Filmdias werden ausgewählt.

**OVERLAY=array(16): <var: char:2>**

Liste der Filmdias (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**OVERLAY=<c-string 2..2: c-string 2-2>**

Filmdia (zugeordnet oder ausgeschlossen) für die aktiven Geräte, über die informiert werden soll.

**FOB=(type, low, high)**

Informationen über aktive Geräte sollen ausgegeben werden,

- für die ein beliebiges FOB-Datendia angegeben wurde (\*ONLY)
- für die kein FOB-Datendia angegeben wurde (\*NO)
- für die FOB-Datendias im spezifizierten Größenbereich angegeben wurden (\*RANGE).

Bei Angabe von \*ANY wirkt der Operand nicht als Auswahlkriterium.

Diese Angaben müssen mit den Einträgen im Operanden FORMS-OVERLAY-BUFFER des Kommandos START-PRINTER-OUTPUT übereinstimmen.

Informationen über Drucker ohne FOB werden nur ausgegeben, wenn FOB=\*ANY angegeben wurde.

*type:* **\*ANY** / **\*ONLY** / **\*NO** / **\*RANGE**

Legt den Typ der Angabe für die Auswahl eines FOB-Datendia fest.

*type:* **\*ANY**

Informationen über Geräte werden unabhängig von FOB-Datendias ausgegeben.

*type:* **\*ONLY**

Informationen über aktive Geräte sollen ausgegeben werden, für die ein beliebiges FOB-Datendia angegeben wurde.

*type:* **\*NO**

Informationen über aktive Geräte sollen ausgegeben werden, für die kein FOB-Datendia angegeben wurde

*type:* **\*RANGE**

Informationen über aktive Geräte sollen ausgegeben werden, für die FOB-Datendias im spezifizierten Größenbereich angegeben wurden

*low:* **\*MIN** / <var: int: 2> / <integer 0..32767>

Legt die Mindestgröße des FOB-Datendias fest. \*MIN entspricht dabei dem Wert 0.

*high:* **\*STD** / <var: int: 2> / <integer 0..32767>

Legt die Maximalgröße des FOB-Datendias fest. \*STD entspricht dabei dem Wert 4032.

**PRIO=(type, low, high)**

Informationen über aktive Geräte, deren Priorität sich im angegebenen Bereich befindet (\*RANGE), sollen ausgegeben werden. Diese Angaben müssen mit den Einträgen im korrespondierenden Operanden des Kommandos START-PRINTER-OUTPUT übereinstimmen.

*type:* **\*ALL** / **\*RANGE**

Art der Drucker, über die Informationen ausgegeben werden sollen.

*type:* **\*ALL**

Informationen über Geräte werden unabhängig von ihrer Priorität ausgegeben.

**type: \*RANGE**

Informationen über Geräte, deren Priorität innerhalb des angegebenen Intervalls liegt, sollen ausgegeben werden.

**low: \*MIN / <var: int: 1> / <integer 30..255>**

Legt den niedrigsten Wert der Priorität fest. \*MIN entspricht dem Wert 30.

**high: \*MAX / <var: int: 1> / <integer 30..255>**

Legt den höchsten Wert der Priorität fest. \*MAX entspricht dem Wert 255.

**CSN=(type, low, high)**

Informationen über aktive Geräte sollen ausgegeben werden,

- die nur einen Zeichensatz benötigen (\*ONE)
- deren Anzahl benötigter Zeichensätze sich im angegebenen Bereich befindet (\*RANGE).

Diese Angaben müssen mit den Einträgen im korrespondierenden Operanden des Kommandos START-PRINTER-OUTPUT übereinstimmen..

Informationen über Nicht-HP-Drucker werden nur ausgegeben, wenn CHARACTER-SET-NUMBER=ALL angegeben wurde.

**type: \*ANY / \*ONE / \*RANGE**

Art der Geräte, über die Informationen ausgegeben werden sollen.

**type: \*ANY**

Informationen über aktive Geräte sollen unabhängig von der Anzahl der Zeichensätze ausgegeben werden.

**type: \*ONE**

Informationen über aktive Geräte, die nur einen Zeichensatz benötigen sollen ausgegeben werden.

**type: \*RANGE**

Informationen über aktive Geräte sollen ausgegeben werden, deren Anzahl benötigter Zeichensätze sich im angegebenen Bereich befindet.

**low: \*MIN/<var: int: 2>/<integer 1..32767>**

Legt die Mindestanzahl an Zeichensätzen fest. \*MIN entspricht dabei dem Wert 1.

**high: \*MAX/<var: int: 2>/<integer 1..32767>**

Legt die Maximalanzahl an Zeichensätzen fest. \*MAX entspricht dabei dem Wert 64.

**SRV\_TYP=\*ALL/<var: enum-of \_flag\_set:1>/\*LIST/\*WILD/\*HOME**

Legt den Typ des Eintrags für die Server-Namen fest.

**SRV\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Server-Namen werden ausgewählt

**SRV\_TYP=<var: enum-of \_flag\_set:1>**

SRV\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**SRV\_TYP=\*LIST**

Liste der Server-Namen für die aktiven Geräte, über die informiert werden soll.

**SRV\_TYP=\*WILD**

Die Server-Namen werden mit Platzhaltern teilqualifiziert angegeben.

**SRV\_TYP=\*HOME**

Die Anforderung wird an den lokalen Server geleitet. Die Drucker können auf verteilter oder lokaler Basis gestartet werden.

**SERVER=\*ALL/array (16): <var: char: 8>/<c-string 1..8: c-string 1-8>**

Das SPOOL-Gerät wird durch den Server, der es verwaltet ausgewählt. Aktive Drucker können Druckaufträge nur ausführen, wenn der Server dies gestattet, der zum Rechner gehört, mit dem der Drucker verbunden ist. Wenn ein entferntes Cluster ausgewählt wurde, ist hier nur der Wert \*ALL erlaubt.

**SERVER=\*ALL**

Vorgabewert für diesen Operanden. Die SPOOL-Geräte werden ausgewählt, die von allen Servern auf allen Rechner im adressierten Cluster verwaltet werden.

**SERVER=array (16): <var: char: 8>**

Die SPOOL-Geräte werden ausgewählt, die von allen spezifizierten Servern auf den Rechner im adressierten Cluster verwaltet werden.

**SERVER=<c-string 1..8: c-string 1-8>**

Die SPOOL-Geräte werden ausgewählt, die von dem spezifizierten Servern auf den Rechner im adressierten Cluster verwaltet werden.

**SRW\_WLD=\*ALL/<var: char:24>/<c-string 1..24: c-string 1-24>**

Die SPOOL-Geräte werden ausgewählt, die von dem spezifizierten Server auf den Rechner im adressierten Cluster verwaltet werden. Die Server werden mit Platzhaltern teilqualifiziert angegeben.

**SRW\_WLD=\*ALL**

Vorgabewert für diesen Operanden. Die SPOOL-Geräte werden ausgewählt, die von allen Servern auf allen Rechner im adressierten Cluster verwaltet werden.

**SRW\_WLD=<var: char: 24>/<c-string 1..24: c-string 1-24>**

Die Server werden mit Platzhaltern teilqualifiziert angegeben.

**HST\_TYP=<var: enum-of \_flag\_set:1> / \*LIST / \*LOCAL\_CLUSTER / \*ALL / \*HOME / \*EXCEPT / \*CLIST**

Legt den Typ des Eintrags für die Namen der Rechner fest.

**HST\_TYP=<var: enum-of \_flag\_set:1>**

HST\_TYP wird nicht direkt, sondern indirekt durch ein Feld mit konstantem Inhalt (equate) ausgewählt. Ein Integerwert gibt dabei den Typ an. Für die Beziehung zwischen Werten und Typen siehe die Tabelle bei DVC\_TYP.

**HST\_TYP=\*LIST**

Liste der Rechnernamen für die aktiven Geräte, über die informiert werden soll.

**HST\_TYP=\*LOCAL\_CLUSTER**

Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge von jedem Rechner im lokalen Cluster ausgeführt werden können.

**HST\_TYP=\*ALL**

Vorgabewert für diesen Operanden. Alle Rechnernamen werden ausgewählt

**HST\_TYP=\*HOME**

Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge vom lokalen Rechner ausgeführt werden können.

**HST\_TYP=\*EXCEPT**

Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge von jedem Rechner verarbeitet werden können, außer von den in dieser Liste aufgeführten Hosts.

**HST\_TYP=\*CLIST**

Alle Elemente der Liste der Host-Namen müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**HSTNAME=\*ALL/array (16): <var: char: 8>/<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive Geräte angefordert, auf denen Druckaufträge von den angegebenen Rechnern ausgeführt oder nicht ausgeführt werden können.

**HSTNAME=\*ALL**

Vorgabewert für diesen Operanden. Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge von allen Rechner von Clustern ausgeführt werden können.

**HSTNAME=array (16): <var: char: 8>**

Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge von den angegebenen Rechnern ausgeführt oder nicht ausgeführt werden können.

**HSTNAME=<c-string 1..8: c-string 1-8>**

Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge von dem angegebenen Rechner ausgeführt oder nicht ausgeführt werden können.

**CLUSTER=\*LOCAL\_CLUSTER/<var: char: 8>/<c-string 1..8: c-string 1-8>**

Das aktive Geräte wird durch den Cluster ausgewählt, der den Server enthält, der das Gerät verwaltet.

**CLUSTER=\*LOCAL CLUSTER**

Vorgabewert für diesen Operanden. Es werden Informationen über aktive Geräte ausgegeben, auf denen Druckaufträge von allen Rechnern im lokalen Cluster ausgeführt werden können.

**CLUSTER=<var: char: 8>**

Das aktive Geräte wird durch den Cluster ausgewählt, der den Server enthält, der das Gerät verwaltet.

**CLUSTER=<c-string 1..8: c-string 1-8>**

Das aktive Geräte wird durch den Cluster ausgewählt, der den Server enthält, der das Gerät verwaltet.

*Hinweis*

Im Rahmen der Interoperabilität zwischen BS2000-Systemen und UNIX-Systemen werden neue Druckerzustände eingeführt, die auf der ISO-Referenz basieren. Die folgenden Zustände werden vom SASDSPO-Makro angezeigt:

- **I** (IDLE): Der Drucker ist druckbereit; es wird kein Druckauftrag gegenwärtig ausgeführt.
- **R** (RUN): Der Drucker ist aktiv.
- **M** (Message): Es gibt eine Konsolmeldung, die den Drucker betrifft.
- **D** (DETACHED): Dies ist entweder der Status zwischen Eintrag im START-PRINTER-OUTPUT-Kommando und seiner Ausführung durch die Controller-Task oder der Status nach Eingabe des STOP-PRINTER-OUTPUT-Kommandos.
- **T** (TRANSFER): Ein gestarteter SPOOL-OUT-Auftrag wartet auf die Transportbestätigung.
- **U** (UNKNOWN): Der Status ist nicht bekannt.

Die Verbindungen zwischen diesen Zuständen und den Zuständen in der ISO-Referenz können aus folgender Tabelle entnommen werden:

Angezeigter Zustand	Druckaufträge für Drucker des Typs			ISO Zustand
	Spool	APA	RSO	
A (ATTACHED)			X	idle
I (IDLE)	X	X	X	idle
T (TRANSFER)	X		X	running
R (RUN)	X	X	X	running
M (MESSAGE)	X	X	X	needs attention
S (STOPPED)			X	needs attention
W (WORKING FOR ADM)			X	needs attention
D (DETACHED)	X	X	X	shutdown
U (UNKNOWN)	X	X	X	shutdown

## Returncodes

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	Kein Fehler
02	00	xxxx	Warnung - Drucker nicht gefunden oder Verarbeitung nicht erlaubt
03	01	FFFF	Pufferlänge zu kurz
01	01	xxxx	Ungültiger DEVICE-Operand
02	01	xxxx	Ungültiger DESTINATION-Operand
03	01	xxxx	Ungültiger FORM-NAME-Operand
04	01	xxxx	Ungültiger OVERLAY-Operand
05	01	xxxx	Ungültiger HOST-NAME-Operand
0A	01	xxxx	Ungültiger USERID-Operand
0B	01	xxxx	Ungültiger SPOOLOUT-CLASS-Operand
0C	01	xxxx	Ungültiger JOB-NAME-Operand
0D	01	xxxx	Ungültiger ACCOUNT-Operand
0E	01	xxxx	Ungültiger ROTATION-Operand
0F	01	xxxx	Ungültiger TWO-UP-Operand
10	01	xxxx	Ungültiger FORM-OVERLAY-BUFFER-Operand
11	01	xxxx	Ungültiger PRIORITY-Operand
12	01	xxxx	Ungültiger CHARACTER-SET-NUMBER-Operand
13	01	xxxx	Ungültige Adresse der RSO-Parameter-Listenerweiterung



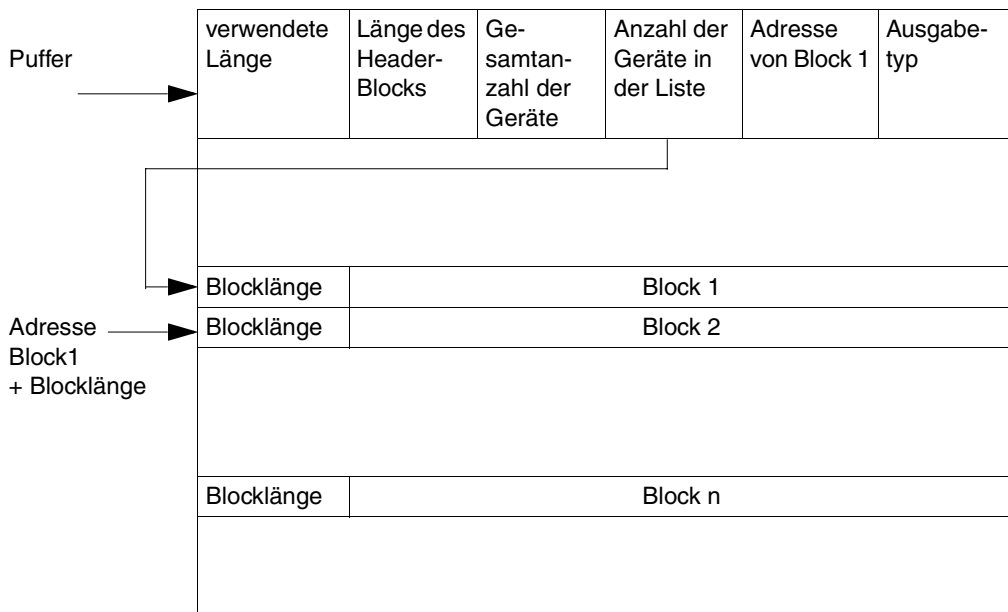
(SC2)	SC1	Maincode	Bedeutung
15	01	xxxx	Ungültiger EXIT -Operand
17	01	xxxx	Ungültige Adresse der Dprint-Parameter-Listenerweiterung
18	01	xxxx	Ungültige Adresse der SPS-Parameter-Listenerweiterung
1A	01	xxxx	Ungültige Adresse der DSEM-Parameter-Listenerweiterung
1B	01	xxxx	Ungültiger BUFFER-Operand
1C	01	xxxx	Ungültiger BUFFER LENGTH-Operand
1D	01	xxxx	Ungültiger SCHEDULING-STATE-Operand
1E	01	xxxx	Ungültiger STATUS-Operand
1F	01	xxxx	Ungültiger CLUSTER-Operand
20	01	xxxx	Ungültiger SERVER-NAME-Operand
21	01	xxxx	Ungültiger ERROR-MESSAGE-Operand
00	20	xxxx	Systemfehler
02	40	xxxx	Keine Zugriffsberechtigung
04	40	xxxx	Operandenwert-Fehler
05	40	xxxx	Die Parameterlisten-Version wird nicht unterstützt
02	80	xxxx	RSO-Subsystem ist nicht geladen
05	80	xxxx	SPS-Subsystem ist nicht geladen
06	80	xxxx	Dprintcl-Subsystem ist nicht geladen
07	80	xxxx	Dprintcm-Subsystem ist nicht geladen
08	80	xxxx	Dprintsv-Subsystem ist nicht geladen
FF	FF	FFFF	SPOOL-Subsystem ist nicht geladen

### *Hinweise*

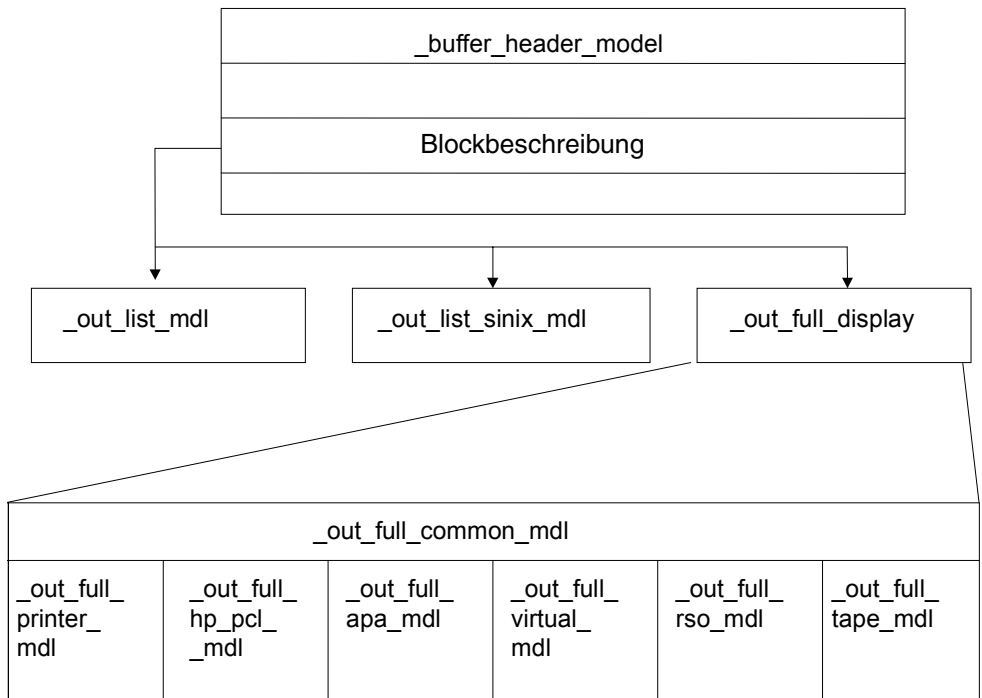
Die vier Zeichen xxxx in der Maincode-Spalte bezeichnen eine Meldungskennung. Um eine ausführlichere Information über einen Fehler zu erhalten, geben Sie das Kommando HELP-MSG SCPxxxx oder SPSxxxx im SYSTEM MODUS ein.

- Das SASDSPO-Makro gibt den obigen Returncode zurück. Der Returncode von HDRCHECK und \$VALID kann zusätzlich zurückgegeben werden. Wenn keine Daten für den Header vorliegen, wird der Wert X'0000003C' im Register 15 zurückgegeben.
- Die Information über das Gerät ist in einem Ausgabepuffer gespeichert, der vom Benutzer bereitgestellt werden muss. Wenn die Pufferlänge zu kurz ist, wird der Makro mit der Fehlermeldung X'03 01 FFFF' "Ausgabebereich zu kurz" beendet. Ein Puffer, der für die gesamte Anzahl der Geräte lang genug ist, muss angefordert werden, um die ganze Information zu erhalten. Ein zusätzlicher Aufruf von SASDSPO gibt dann die gewünschte Information aus.

- Die Sektion `_output_buffer_model` wird immer in den Puffer geschrieben. Sie enthält die genaue Zahl der aktiven SPOOL-Geräte, die Adresse des ersten Blocks, der die Informationen über ein aktives Gerät enthält und die Länge des betreffenden Blocks sowie den Ausgabebetyp. Die Adresse jedes zusätzlichen Informationsblocks kann aus der Adresse des ersten Informationsblocks und seiner Länge errechnet werden.



- Die Blockbeschreibung verändert sich je nach Ausgabebetyp in Länge und Layout. Das gesamte Anzeigenlayout besteht aus einem allgemeinen Teil und einem Teil, der sich auf den Gerätetyp bezieht.



- Informationen über das RSO-Subsystem werden von RSO selbst bereitgestellt, wenn dieses Subsystem geladen ist. Allerdings betreffen diese Informationen nur Geräte, die von RSO verwaltet werden.
- Der Aufrufer kann in der Eingabeparameterliste die Adresse des von ihm eigens reservierten Puffers genauso wie die korrespondierende Länge mit Hilfe der Operanden BUFFER und BUFLen angeben. In diesem Fall kann während des Druckprozesses eine Warnung zurückgegeben werden, dass der bereitgestellte Puffer zu kurz ist. Da sowohl die gesamte Anzahl der Drucker als auch die Anzahl der tatsächlich im Puffer zurückgemeldeten Drucker im Puffer-Header angegeben wird, kann der Benutzer die benötigte Pufferlänge berechnen und die Schnittstelle erneut aufrufen. Der Aufrufer hat auch die Möglichkeit die Schnittstelle mit BUFFER=\*NONE und BUFLen=\*NONE (Standardwerte) aufzurufen. In diesem Fall fordert der Prozess selbst einen Puffer mit geeigneter Größe im Klasse-6-Speicher an. Nach dem Aufruf wird der Pufferzeiger und seine Länge in die Eingabeparameterliste geschrieben. Der Aufrufer muss den Pufferspeicher freigeben.

**Beschreibung der Parameterliste**

```

*   parameter list description
SSS_MDL          DS    0F
SSSHDR  FHDR MF=(C,SSS),EQUATES=NO      Standard header
*   main return codes
AIF  ('&EQUATES' NE 'YES').Q01001
SSSRCNO          EQU    0      subcode1 = 00 : no error
SSSWARN          EQU    2      job not found or processing
*               not permitted
SSSVADM          EQU    2      subcode1 = 01 : pl area
*               invalid - maincode = FFFF
SSSPARM          EQU    1      subcode1 = 01 : wrong
*               parameter
SSSDEVI          EQU    1      subcode1 = 01 : device
SSSDEST          EQU    2      subcode1 = 01 : destination
SSSFONA          EQU    3      subcode1 = 01 : form name
SSSOVRL          EQU    4      subcode1 = 01 : overlay
SSSHOST          EQU    5      subcode1 = 01 : host
SSSPPOU          EQU    6      subcode1 = 01 : page printer
*               output
SSSPRSA          EQU    7      subcode1 = 01 : print sample
SSSTRAE          EQU    8      subcode1 = 01 : trace
SSSACCE          EQU    9      subcode1 = 01 : access
SSSUSER          EQU   10      subcode1 = 01 : userid
SSSPCL           EQU   11      subcode1 = 01 : spoolout
*               class
SSSJNAM          EQU   12      subcode1 = 01 : job name
SSSACCN          EQU   13      subcode1 = 01 : account
SSSROTT          EQU   14      subcode1 = 01 : rotation
SSSTWOU          EQU   15      subcode1 = 01 : two-up
SSSFOB           EQU   16      subcode1 = 01 : fob
SSSPRIO          EQU   17      subcode1 = 01 : priority
SSSCSNU          EQU   18      subcode1 = 01 : char_set_num
SSRSOL           EQU   19      subcode1 = 01 : addr of rso
*               pl
SSSREVI          EQU   20      subcode1 = 01 : revision
SSSEXTT          EQU   21      subcode1 = 01 : exit
SSSDPPL          EQU   23      subcode1 = 01 : addr of
*               dprintpl
SSSSPSL          EQU   24      subcode1 = 01 : addr of sps
*               pl
SSSSCSL          EQU   25      subcode1 = 01 : addr of scs
*               pl
SSSDSEM          EQU   26      subcode1 = 01 : addr of dsem
*               pl
SSSBUFF          EQU   27      subcode1 = 01 : addr of
*               buffer

```

```

SSSBULE          EQU  28          subcode1 = 01 : buffer length
SSSSCHE          EQU  29          subcode1 = 01 : scheduling
*               state
SSSSTAU          EQU  30          subcode1 = 01 : status
SSSCLUT          EQU  31          subcode1 = 01 : cluster
SSSSERN          EQU  32          subcode1 = 01 : server
SSSERM           EQU  33          subcode1 = 01 : error message
SSSEINF          EQU  34          subcode1 = 01 : information
SSSSHRT          EQU   3          output area too short - main
*               code=FFFF
SSSSYSE          EQU   0          subcode1 = 20 : system error
SSSPRIV          EQU   2          subcode1 = 40 : no
*               authorization
SSSVERR          EQU   4          operand value error
SSSVERE          EQU   5          p/l version not supported
SSSNOSS          EQU   0          subcode1 = 80 : subsystem not
*               loaded
SSSNRSO          EQU   2          rso not loaded
SSSNDCL          EQU   6          dprintcl not loaded
SSSNDCM          EQU   7          dprintcm not loaded
SSSNDSV          EQU   8          dprintsv not loaded
SSSNSPO          EQU  255         subcode1 = 255: spool not
*               loaded
.Q01001  ANOP
*
SSSDPCP          DS    A          address of dprint_pl
SSSPSP           DS    A          address of sps_pl
SSRSOP           DS    A          address of rso_pl
SSSSCSP          DS    A          address of scs_pl
SSSDSEP          DS    A          address of dsem_pl
SSSRES5          DS    A
SSSRES6          DS    A
SSSRES7          DS    A
SSSBUF@         DS    A          Address of a user-specified
*               area where the output must
*               be returned.
SSSBUFL          DS    F          user buffer length
SSSSCHS          DS    FL1        scheduling state
*   _sched_state_set
AIF ('&EQUATES' NE 'YES').Q01002
SSSSSNE          EQU   2          scheduling state = *next
SSSSSCU          EQU   1          scheduling state = *current
.Q01002  ANOP
*
SSSROTA          DS    FL1        rotation
*   rotation_set
AIF ('&EQUATES' NE 'YES').Q01003
SSSROSA          EQU   1          rotation = *any
    
```

```

SSRSOSY          EQU  2          rotation = *yes
SSRSOSN          EQU  4          rotation = *no
SSRSOSM          EQU  8          rotation = *manual
.Q01003  ANOP
*
SSSTWUP          DS    FL1          two-up
*  two_up_set
AIF  ('&EQUATES' NE 'YES').Q01004
SSSTUPA          EQU  131         two-up = *any
SSSTUPY          EQU  3          two-up = *yes
SSSTUPN          EQU  128        two-up = *no
SSSTUM1          EQU  1          two-up = *mode-1
SSSTUM2          EQU  2          two-up = *mode-2
SSSTUDF          EQU  8          two-up = not specified
.Q01004  ANOP
*
SSSEXIT          DS    FL1          exit
*  exit_set
AIF  ('&EQUATES' NE 'YES').Q01005
SSSEXSA          EQU  4          exit = *any
SSSEXSY          EQU  1          exit = *yes
SSSEXSN          EQU  0          exit = *no
.Q01005  ANOP
*
SSSSTAT          DS    FL1          printer status
*  _printer_status_set
AIF  ('&EQUATES' NE 'YES').Q01006
SSSPSAT          EQU  193         printer state = *attached
SSSPSID          EQU  201         printer state = *idle
SSSPSTR          EQU  227         printer state = *transfer
SSSPSRU          EQU  217         printer state = *run
SSSPSME          EQU  212         printer state = *message
SSSPSST          EQU  226         printer state = *stopped
SSSPSWO          EQU  230         printer state = *worked
SSSPSDE          EQU  196         printer state = *detached
SSSPSUN          EQU  228         printer state = *unknown
SSSPSAL          EQU  1          printer state = *all
.Q01006  ANOP
*
SSSSINF          DS    FL1          information
*  _information_set
AIF  ('&EQUATES' NE 'YES').Q01007
SSSSINS          EQU  0          information = *summary
SSSSINA          EQU  1          information = *all
*.Q01007  ANOP
*
SSSUN2           DS    XL2
SSSCLUS          DS    CL8          cluster

```

```

*
SSSADEVN          DS    0XL156      device
SSSAFC8           DS    FL1
**  _flag_set
AIF  ('&EQUATES' NE 'YES').Q01008
SSSFALL          EQU    0           _type = *all
SSSFNON          EQU    1           _type = *none
SSSFLOC          EQU    2           _type = *local
SSSFREM          EQU    3           _type = *remote
SSSFDEV          EQU    5           _type = *public_remote
SSSFCLCL         EQU    6           _type = *local_cluster
SSSFWLD          EQU    7           _type = *wildcard
SSSFST           EQU    8           _type = *list
SSSFSTD          EQU    9           _type = *std
SSSFHOM          EQU    10          _type = *home
SSSFONL          EQU    11          _type = *only
SSSFRNG          EQU    12          _type = *range
SSSFONE          EQU    13          _type = *one
SSSFEXC          EQU    14          _type= *except
SSSFANY          EQU    15          _type= *any
SSSFNO           EQU    16          _type= *no
.Q01008 ANOP
*
SSSAWC8           DS    CL24
SSSAUC8           DS    XL3
SSSAEC8           DS    8CL8
AIF  ('&EQUATES' NE 'YES').Q01009
SSSAEC8#          EQU    8
.Q01009 ANOP
SSSAEU8           DS    8CL8
AIF  ('&EQUATES' NE 'YES').Q01010
SSSAEU8#          EQU    8
.Q01010 ANOP
*
*
SSSBDEST          DS    0XL156      destination
SSSBFC8           DS    FL1
SSSBWC8           DS    CL24
SSSBUC8           DS    XL3
SSSBEC8           DS    16CL8
AIF  ('&EQUATES' NE 'YES').Q01011
SSSBEC8#          EQU    16
.Q01011 ANOP
*
*
SSSCSRVN          DS    0XL156      server name
SSSCFC8           DS    FL1
SSSCWC8           DS    CL24
    
```

```

SSSCUC8          DS    XL3
SSSCEC8          DS    16CL8
                AIF    ('&EQUATES' NE 'YES').Q01012
SSSCEC8#         EQU    16
.Q01012 ANOP
*
*
SSSDHOST         DS    0XL156      host name
SSSDFC8          DS    FL1
SSSDWC8          DS    CL24
SSSDUC8          DS    XL3
SSSDEC8          DS    16CL8
                AIF    ('&EQUATES' NE 'YES').Q01013
SSSDEC8#         EQU    16
.Q01013 ANOP
*
*
SSSEMSG          DS    0XL156      error messae
SSSEFC8          DS    FL1
SSSEWC8          DS    CL24
SSSEUC8          DS    XL3
SSSEEC8          DS    1CL8
                AIF    ('&EQUATES' NE 'YES').Q01014
SSSEEC8#         EQU    1
.Q01014 ANOP
SSSEEU8          DS    15CL8
                AIF    ('&EQUATES' NE 'YES').Q01015
SSSEEU8#         EQU    15
.Q01015 ANOP
*
*
SSSFFORM         DS    0XL124      form name
SSSFFC6          DS    FL1
SSSFWC6          DS    CL24
SSSFUC6          DS    XL3
SSSFEC6          DS    16CL6
                AIF    ('&EQUATES' NE 'YES').Q01016
SSSFEC6#         EQU    16
.Q01016 ANOP
*
*
SSSGUSER         DS    0XL156      userid
SSSGFC8          DS    FL1
SSSGWC8          DS    CL24
SSSGUC8          DS    XL3
SSSGEC8          DS    16CL8
                AIF    ('&EQUATES' NE 'YES').Q01017
SSSGEC8#         EQU    16

```



```

.Q01017 ANOP
*
*
SSSHSCLA          DS    0XL20      spoolout class
SSSHFI1           DS    FL1
SSSHUI2           DS    XL3
SSSHEI1           DS    16X
*
*
SSSISNAM          DS    0XL156     job name
SSSIFC8           DS    FL1
SSSIWC8           DS    CL24
SSSIUC8           DS    XL3
SSSIEC8           DS    16CL8
AIF ('&EQUATES' NE 'YES').Q01018
SSSIEC8#          EQU    16
.Q01018 ANOP
*
*
SSSJACCO          DS    0XL156     account
SSSJFC8           DS    FL1
SSSJWC8           DS    CL24
SSSJUC8           DS    XL3
SSSJEC8           DS    16CL8
AIF ('&EQUATES' NE 'YES').Q01019
SSSJEC8#          EQU    16
.Q01019 ANOP
*
*
SSSKOVER          DS    0XL60      overlay
SSSKFC2           DS    FL1
SSSKWC2           DS    CL24
SSSKUC2           DS    XL3
SSSKEC2           DS    16CL2
AIF ('&EQUATES' NE 'YES').Q01020
SSSKEC2#          EQU    16
.Q01020 ANOP
*
*
SSSLF0B           DS    0XL8       fob
SSSLFR2           DS    FL1
SSSLUN1           DS    XL1
SSLLR2            DS    H
SSSLHR2           DS    H
SSSLUR2           DS    XL2
*
*
SSSMCSN           DS    0XL8       char set number
    
```

```

SSSMFR2          DS    FL1
SSSMUN1          DS    XL1
SSSMLR2          DS    H
SSSMHR2          DS    H
SSSMUR2          DS    XL2
*
*
SSSNPRIO         DS    0XL4      priority
SSSNFR1          DS    FL1
SSSNLR1          DS    X
SSSNHR1          DS    X
SSSNUR1          DS    XL1
*
SSSUN6           DS    XL16
      AIF    ('&EQUATES' NE 'YES').Q01021
SSS#             EQU    *-SSSHDR
.Q01021 ANOP
*   _out_md1
SSS_OMDL         DS    0F
*
SSSBHDR          DS    0XL24      buffer header
SSSHUSL          DS    F          total used length
SSSHBL          DS    F          header block length
SSSHTDV          DS    F          total dvc count
SSSHRDV          DS    F          return dvc count
SSSHFBD          DS    F          displacement to first block
SSSHOTY          DS    FL1       output type
*   _output_type_set
      AIF    ('&EQUATES' NE 'YES').Q01022
SSSOTLI          EQU    1          output type = list
SSSOTFU          EQU    2          output type = full
SSSOTSI          EQU    4          output type = sinix
.Q01022 ANOP
*
SSSHDU2          DS    XL3
*
SSS_DISPLAY_ENTRY DS    0XL1628   _display_entry
*
SSSFULD          DS    0XL1628   full display
*
SSSFCOM          DS    0XL1416   common part
SSSFCBL          DS    H          block length
SSSFCDN          DS    CL8       device name
SSSFCDT          DS    CL16      device type
SSSFCUI          DS    CL8       userid of the current job
SSSFACTS         DS    CL4       tsn of the current job
SSSFCLP          DS    X          range of priority : low value
SSSFCHP          DS    X          range of priority : high

```

```

*
SSSF CRV          DS    X          value
SSSF UNOV        DS    X          revision number
SSSF FCUS        DS    FL1        type of device activity
*   _out_use_set
      AIF ('&EQUATES' NE 'YES').Q01023
SSSF INU          EQU    1          use = input
SSSF PGU          EQU    2          use = page printer
SSSF NOU          EQU    4          use = no
SSSF OUU          EQU    8          use = output
.Q01023 ANOP
*
SSSF CSS         DS    FL1        scheduling state
*   _out_sched_state_set
      AIF ('&EQUATES' NE 'YES').Q01024
SSSF SCC          EQU    1          scheduling state = current
SSSF SCN          EQU    2          scheduling state = next
.Q01024 ANOP
*
SSSF CAI         DS    FL1        additional info type
*   _add_info_type_set
      AIF ('&EQUATES' NE 'YES').Q01025
SSSF I LN        EQU    0          info = no
SSSF I LI        EQU    1          info = line printer
SSSF I HP        EQU    2          info = hp/hp90 printer
SSSF I AP        EQU    3          info = apa printer
SSSF I VI        EQU    4          info = virtual printer
SSSF I RS        EQU    5          info = rso printer
SSSF I TA        EQU    6          info = tape printer
.Q01025 ANOP
*
SSSF FUSD        DS    XL3
*
SSSF PCLASS      DS    0XL20      class
SSSF FCCN        DS    H          number of elements in list
SSSF FCCI        DS    FL1        indicator
*   _out_indicator_set
      AIF ('&EQUATES' NE 'YES').Q01026
SSSF INZ         EQU    0          no value set
SSSF INE         EQU    1          negative list
SSSF IPO         EQU    16         positive list
SSSF IAL         EQU    20         all or any
.Q01026 ANOP
*
SSSF FCCU        DS    XL1
SSSF FCLL        DS    16X        list of spoolout class
*
*
```

```

SSSF0RM          DS    0XL100      form
SSSF0CFN        DS    H            number of elements in list
SSSF0CFI        DS    FL1         indicator
*   _out_ind_form_set
      AIF  ('&EQUATES' NE 'YES').Q01027
SSSF0FNG        EQU    1          negative list
SSSF0FNE        EQU    19         negative equivalent
SSSF0FAL        EQU    4          all
SSSF0FAE        EQU    22         all equivalent
SSSF0FPO        EQU    16         positive list
SSSF0FPE        EQU    18         positive equivalent
.Q01027  ANOP
*
SSSF0CFU        DS    XL1
SSSF0CFE        DS    16CL6       list of form name
      AIF  ('&EQUATES' NE 'YES').Q01028
SSSF0CFE#       EQU    16
.Q01028  ANOP
*
*
SSSF0PNAME      DS    0XL132      pname
SSSF0FCPN       DS    H            number of elements in list
SSSF0FCPI       DS    FL1         indicator
SSSF0FCPU       DS    XL1
SSSF0FCPE       DS    16CL8       list of job name
      AIF  ('&EQUATES' NE 'YES').Q01029
SSSF0FCPE#      EQU    16
.Q01029  ANOP
*
*
SSSF0USERID     DS    0XL132      userid
SSSF0FCUN       DS    H            number of elements in list
SSSF0FCUD       DS    FL1         indicator
SSSF0FCUU       DS    XL1
SSSF0FCUE       DS    16CL8       list of user ids
      AIF  ('&EQUATES' NE 'YES').Q01030
SSSF0FCUE#      EQU    16
.Q01030  ANOP
*
*
SSSF0ACCOUNT    DS    0XL132      account
SSSF0FCAN       DS    H            number of elements in list
SSSF0FCAD       DS    FL1         indicator
SSSF0FCAU       DS    XL1
SSSF0FCAE       DS    16CL8       list of account number
      AIF  ('&EQUATES' NE 'YES').Q01031
SSSF0FCAE#      EQU    16
.Q01031  ANOP

```

```

*
*
SSSDESTIN          DS    0XL812    destin
SSSFCLN            DS    H          number of elements in list
SSSFCDI            DS    FL1        indicator
SSSFCDU            DS    XL1
SSSFCDCE           DS    101CL8     list of device pools
      AIF    ('&EQUATES' NE 'YES').Q01032
SSSFCDCE#          EQU    101
.Q01032 ANOP
*
SSSFCD A           DS    CL10       last scheduling date
SSSFSTI            DS    CL5        last scheduling time
SSSFSE             DS    CL1        season (S/W)
SSSFEMS            DS    CL8        error msg
SSSFRSX            DS    XL16
*
SSS_OUT_FULL_DEDICATED DS    0XL212    _out_full_dedicated
*
SSSFHPC            DS    0XL212
SSSFHBL            DS    H          block length
SSSFHFN            DS    CL6        current form used
SSSFHSN            DS    CL8        server name
SSSFHHN            DS    CL8        host name
SSSFHLF            DS    H          range of fob : low value
SSSFHHF            DS    H          range of fob : high value
SSSFHLC            DS    H          range of char set num : low
*                               value
SSSFHHC            DS    H          range of char set num : high
*                               value
SSSFHTR            DS    FL1        trace file to be generated
*   _yes_no_set
      AIF    ('&EQUATES' NE 'YES').Q01033
SSSSTYE            EQU    1
SSSSTNO            EQU    2
.Q01033 ANOP
*
SSSFHEX            DS    FL1        exit routines activated
SSSFHSA            DS    FL1        sample printout
SSSFHST            DS    FL1        printer status
*   _out_prt_status_set
      AIF    ('&EQUATES' NE 'YES').Q01034
SSSOPSA            EQU    193       C'A'
SSSOPSI            EQU    201       C'I'
SSSOPST            EQU    227       C'T'
SSSOPSR            EQU    217       C'R'
SSSOPSM            EQU    212       C'M'
SSSOPSS            EQU    226       C'S'

```

```

SSSOPSW          EQU    230          C'W'
SSSOPSD          EQU    196          C'D'
SSSOPSU          EQU    228          C'U'
.Q01034 ANOP
*
SSSFHSC          DS      FL1          printer scope
*  _scope_set
    AIF  ('&EQUATES' NE 'YES').Q01035
SSSSCNO          EQU     0           scope = no
SSSSCLO          EQU     1           scope = local
SSSSCDI          EQU     2           scope = distributed
.Q01035 ANOP
*
SSSFHET          DS      FL1          class of error
*  _out_der_type_set
    AIF  ('&EQUATES' NE 'YES').Q01036
SSSDERN          EQU     0           device erro type = no
SSSDERO          EQU     1           device erro type = oper
SSSDERD          EQU     2           device erro type = data
SSSDERL          EQU     4           device erro type = load
SSSDERJ          EQU     8           device erro type = job
.Q01036 ANOP
*
SSSFHTU          DS      FL1          two-up processing
*  _out_two_up_set
    AIF  ('&EQUATES' NE 'YES').Q01037
SSSTUM1          EQU     1           two-up = mode-1
SSSTUM2          EQU     2           two-up = mode-2
SSSTUPY          EQU     7           two-up = yes
SSSTUPN          EQU    128          two-up = no
SSSTUPA          EQU    135          two-up = any
.Q01037 ANOP
*
SSSFHRO          DS      FL1          rotation
*  _out_rotation_set
    AIF  ('&EQUATES' NE 'YES').Q01038
SSSSRAN          EQU     1           rotation = *any
SSSSRYE          EQU     2           rotation = *yes
SSSSRNO          EQU     4           rotation = *no
SSSSRMA          EQU     8           rotation = *manual
.Q01038 ANOP
*
SSSFHCD          DS      CL2          current dia
*
SSSHOST_HP_PCL   DS      0XL132      host_hp_pcl
SSSFHLN          DS      H           number of elements in list
SSSFHHI          DS      FL1          indicator
*  _out_ind_host_set

```

```

        AIF ('&EQUATES' NE 'YES').Q01039
SSSSHPO          EQU 16          positive
SSSSHNG          EQU 1          negative
SSSSHAL          EQU 20         all_cluster
SSSSHHO          EQU 48         home
SSSHLO           EQU 80         local
.Q01039 ANOP
*
SSSFHHU          DS XL1
SSSFHHE          DS 16CL8       list of host name
        AIF ('&EQUATES' NE 'YES').Q01040
SSSFHHE#         EQU 16
.Q01040 ANOP
*
*
SSSDIA_HP_PCL   DS 0XL36       dia_hp_pcl
SSSFHDN          DS H          number of elements in list
SSSFHDI          DS FL1        indicator
*   _out_ind_dia_set
        AIF ('&EQUATES' NE 'YES').Q01041
SSSSDPO          EQU 16         positive
SSSSDNG          EQU 1         negative
SSSSDAL          EQU 20         all
SSSSDNO          EQU 32        none
SSSSDON          EQU 64         only
.Q01041 ANOP
*
SSSFHDU          DS XL1
SSSFHDE          DS 16CL2       list of overlay
        AIF ('&EQUATES' NE 'YES').Q01042
SSSFHDE#         EQU 16
.Q01042 ANOP
*
*
SSSFUS2          DS XL2
*
        ORG SSS_OUT_FULL_DEDICATED
*
SSSFLIN          DS 0XL164
SSSFLBL          DS H          block length
SSSFLFN          DS CL6        current form used
SSSFLSN          DS CL8        name of the server
SSSFLHN          DS CL8        name of the host
SSSFLTR          DS FL1        trace file to be generated
SSSFLEX          DS FL1        exit routines activated
SSSFLSA          DS FL1        sample printout
SSSFLST          DS FL1        printer status
SSSFLSC          DS FL1        printer scope :
*                                local/distributed

```

```

SSSFLET          DS    FL1          class of error
*
SSSHOST_PRINTER DS    0XL132       host_printer
SSSFLLN         DS    H            number of elements in list
SSSFLHI        DS    FL1          indicator
SSSFLHU        DS    XL1
SSSFLHE        DS    16CL8       list of hostname
      AIF    ('&EQUATES' NE 'YES').Q01043
SSSFLHE#       EQU    16
.Q01043 ANOP
*
SSSFUS1        DS    XL2
*
      ORG    SSS_OUT_FULL_DEDICATED
*
SSSFAPA        DS    0XL164
SSSFABL        DS    H            block length
SSSFABN        DS    CL6         current form used
SSSFASN        DS    CL8         server name
SSSFAHN        DS    CL8         host name
SSSFATR        DS    FL1         trace file to be generated
SSSFALV        DS    FL1         trace file to be generated
*  _out_level_set
      AIF    ('&EQUATES' NE 'YES').Q01044
SSSLEVN        EQU    40         trace level = not spec.
SSSLEVS        EQU    226        trace level = status
SSSLEVB        EQU    194        trace level = blk-cont.
SSSLEVC        EQU    195        trace level = complete
.Q01044 ANOP
*
SSSFAEX        DS    FL1         exit routines activated
SSSFAST        DS    FL1         printer status
SSSFASC        DS    FL1         printer scope
SSSFAET        DS    FL1         class of error
*
SSSHOST_APA    DS    0XL132       host_apa
SSSFALN        DS    H            number of elements in list
SSSFAHI        DS    FL1         indicator
SSSFAHU        DS    XL1
SSSFAHE        DS    16CL8       list of host name
      AIF    ('&EQUATES' NE 'YES').Q01045
SSSFAHE#       EQU    16
.Q01045 ANOP
*
SSSFAUN        DS    XL2
*
      ORG    SSS_OUT_FULL_DEDICATED
*

```



```

SSSFVIR          DS      0XL160
SSSFVBL          DS      H          block length
SSSFVSN          DS      CL8        server name
SSSFVHN          DS      CL8        host name
SSSFVPN          DS      CL4        program task
SSSFVST          DS      FL1        printer status
SSSFVSC          DS      FL1        printer scope
SSSFVET          DS      FL1        class of error
SSSFVU1          DS      XL1
*
SSSHOST_VIRT     DS      0XL132     host_virt
SSSFVLN          DS      H          number of elements in list
SSSFVHI          DS      FL1        indicator
SSSFVHU          DS      XL1
SSSFVHE          DS      16CL8      list of host name
AIF      ('&EQUATES' NE 'YES').Q01046
SSSFVHE#         EQU      16
.Q01046 ANOP
*
SSSFVU2          DS      XL2
*
ORG      SSS_OUT_FULL_DEDICATED
*
SSSFRSO          DS      0XL82
SSSFRBL          DS      H          block length
SSSFRFN          DS      CL6        current form used
SSSFRTR          DS      FL1        trace file to be generated
SSSFREX          DS      FL1        exit routines activated
SSSFRSA          DS      FL1        sample printout
SSSFRST          DS      FL1        printer status
SSSFRHE          DS      4CL4      list of accesses
AIF      ('&EQUATES' NE 'YES').Q01047
SSSFRHE#         EQU      4
.Q01047 ANOP
SSSFJVN          DS      CL54      MONJV name
*
ORG      SSS_OUT_FULL_DEDICATED
*
SSSFTAP          DS      0XL182
SSSFTBL          DS      H          block length
SSSFTLF          DS      H          range of fob : low value
SSSFTHF          DS      H          range of fob : high value
SSSFTLC          DS      H          range of char set num : low
*                               value
SSSFTHC          DS      H          range of char set num : high
*                               value
SSSFTRP          DS      H          retention period
SSSFTIM          DS      CL6        vsn of directory tape
    
```

```

SSSFTTD          DS    CL8          density of tape
SSSFTRM          DS    FL1          processing mode
*   _out_rmode_set
      AIF    ('&EQUATES' NE 'YES').Q01048
SSSSRMZ          EQU    0          no value set
SSSSRMD          EQU    2          rmode = direct
SSSSRMC          EQU    8          rmode = copy
.Q01048 ANOP
*
SSSFTTU          DS    FL1          two-up processing
SSSFTR0          DS    FL1          rotation
SSSFTTY          DS    FL1          selection type
*   _tape_type_flag_set
      AIF    ('&EQUATES' NE 'YES').Q01049
SSSTTFY          EQU    1          type = may
SSSTTFM          EQU    2          type = must
.Q01049 ANOP
*
SSSFTVA          DS    14X         list of type
*
SSSDIA_TAPE      DS    0XL36       dia_tape
SSSFTDN          DS    H           number of elements in list
SSSFTDI          DS    FL1         indicator
SSSFTDU          DS    XL1         indicator
SSSFTDE          DS    16CL2       list of overlay
      AIF    ('&EQUATES' NE 'YES').Q01050
SSSFTDE#         EQU    16
.Q01050 ANOP
*
*
SSSVSN           DS    0XL100      vsn
SSSFTVN          DS    H           number of elements in list
SSSFTVI          DS    FL1         indicator
SSSFTVU          DS    XL1         indicator
SSSFTVE          DS    16CL6       list of vsn
      AIF    ('&EQUATES' NE 'YES').Q01051
SSSFTVE#         EQU    16
.Q01051 ANOP
*
*
SSSFTU2          DS    XL2
*
      ORG    SSS_OUT_FULL_DEDICATED+212
*
      ORG    SSS_DISPLAY_ENTRY
*
SSSFLID          DS    0XL92       list display
SSSLBLK          DS    H           block length
SSSLDVN          DS    CL8         device name

```

```

SSSLDVT          DS    CL16      device type
SSSLUID          DS    CL8        userid of the current job
SSSLTSN         DS    CL4        tsn of the current job
SSSLFRM         DS    CL6        form of the current job
SSSLSPC         DS    X          spoolout class of the current
* job
SSRSOC          DS    X
SSSLEXI         DS    FL1        exit routine activated
SSSLSTA         DS    FL1        printer status
SSSLSCO         DS    FL1        printer scope :
* local/distributed
SSSLSTP         DS    FL1        stop command issued
SSSLCAL         DS    FL1        caller is administrator
SSSLCRI         DS    FL1        explicite criteria given
SSSLCDA         DS    CL10       last scheduling date
SSSLSTI         DS    CL5        last scheduling time
SSLSSE         DS    CL1        season (S/W)
SSSLEMS         DS    CL8        error msg
SSSLRSX         DS    XL16
*
                ORG    SSS_DISPLAY_ENTRY
*
SSSFSLI         DS    0XL18      sinix display
SSLSBL         DS    H          block length
SSLSFN         DS    CL6        current form used
SSLSDN         DS    CL8        device name
SSLSST         DS    FL1        printer status
SSLSUN         DS    XL1
*
                ORG    SSS_DISPLAY_ENTRY+1628
                AIF    ('&EQUATES' NE 'YES').Q01052
SSSOUTPUT#     EQU    *-SSSHUSL
.Q01052 ANOP
    
```

## STPSPPR - Druckerzuweisung aufheben

**Benutzergruppe:** Systembetreuer, RSO-Geräteverwalter

**Programmiersprache:** Assembler, CPP, C

**Makrotyp:** M, S

Mit dem Makro STPSPPR werden Druckerzuweisungen aufgehoben.

### Format

Operation	Operanden
STPSPPR	ENTRY = <u>YES</u> / NO  VARIANT = <c-string-without-quotes 3..3> / <u>001</u>  DVNAME = array(8): <var: char:8> / <c-string 1..8>  FORCE = <u>*STD</u> / *IMMEDIATE

### Operandenbeschreibung

#### **ENTRY=YES/NO**

Angabe zu korrespondierenden Einträgen

#### **ENTRY=YES**

Korrespondierende Einträge werden generiert.

#### **ENTRY=NO**

Korrespondierende Einträge werden nicht generiert.

#### **VARIANT=001 / <c-string-without-quotes 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

#### **DVNAME=list-poss(8): <alphanum-name 1..8> /identifier**

Namen der Drucker, die weggeschaltet werden sollen.

#### **FORCE=\*STD / \*IMMEDIATE**

Bestimmt, ob ein gegenwärtig aktiver Druckauftrag sofort abgebrochen (\*IMMEDIATE) oder vollständig abgearbeitet wird (\*STD).

**Returncode**

<b>(SC2)</b>	<b>SC1</b>	<b>Maincode</b>	<b>Bedeutung</b>
00	00	0000	Kein Fehler
02	00	xxxx	Warnung - Drucker existiert nicht oder Verarbeitung ist nicht erlaubt
02	01	FFFF	Ungültige Parameterlistarea
01	01	xxxx	Ungültiger DEVICE-Operand
16	01	xxxx	Ungültiger FORCE-Operand
00	20	xxxx	Systemfehler
02	40	xxxx	Keine Zugriffsberechtigung
04	40	xxxx	Operandenwert-Fehler
05	40	xxxx	Parameterlist-Version wird nicht unterstützt
FF	FF	FFFF	SPOOL-Subsystem ist nicht geladen

*Hinweis*

Die vier Zeichen xxxx in der Maincode-Spalte bezeichnen eine Meldungskennung. Um eine ausführlichere Information über einen Fehler zu erhalten, geben Sie das Kommando HELP-MSG SCPxxxx oder SPSxxxx im SYSTEM MODUS ein.

**Beschreibung der Parameterliste**

\* parameter list description

STP\_MDL DS OF

STPHDR FHDR MF=(C,STP),EQUATES=NO Standard header

\* main return codes

AIF ('&EQUATES' NE 'YES').Q01001

STPRCNO EQU 0 subcode1 = 00 : no error

STPPARM EQU 1 subcode1 = 01 : wrong

\* parameter

STPSYSE EQU 2 subcode1 = 20 : system error

STPPRIV EQU 2 subcode1 = 40 : no

\* authorization

STPVERR EQU 4 operand value error

STPVERE EQU 5 p/l version not supported

STPNOSS EQU 0 subcode1 = 80 : subsystem not

\* loaded

STPNSPO EQU 1 spool not loaded

STPNRSO EQU 2 rso not loaded

STPDPPL EQU 23 subcode1 = 01 : addr of

\* dprintpl

STPSPSL EQU 24 subcode1 = 01 : addr of sps

\* pl

STPRSOL EQU 19 subcode1 = 01 : addr of rso

\* pl

STPSCSL EQU 25 subcode1 = 01 : addr of scs

\* pl

STPDSEM EQU 26 subcode1 = 01 : addr of dsem

\* pl

STPBUFF EQU 27 subcode1 = 01 : addr of

\* buffer

STPBUFL EQU 28 subcode1 = 01 : buffer length

STPSCHE EQU 29 subcode1 = 01 : scheduling

\* state

STPROTA EQU 14 subcode1 = 01 : rotation

STPTWOU EQU 15 subcode1 = 01 : two-up

STPEXTT EQU 21 subcode1 = 01 : exit

STPSTAT EQU 30 subcode1 = 01 : status

STPCLUS EQU 31 subcode1 = 01 : cluster

STPDEVI EQU 1 subcode1 = 01 : device

STPDEST EQU 2 subcode1 = 01 : destination

STPSERN EQU 32 subcode1 = 01 : server

STPHOST EQU 5 subcode1 = 01 : host

STPERRM EQU 33 subcode1 = 01 : error message

STPFONA EQU 3 subcode1 = 01 : form name

STPUSER EQU 10 subcode1 = 01 : userid

STPSPCL EQU 11 subcode1 = 01 : spoolout

\* class

STPJNAM	EQU	12	subcode1 = 01 : job name
STPACCN	EQU	13	subcode1 = 01 : account
STPOVRL	EQU	4	subcode1 = 01 : overlay
STPFOB	EQU	16	subcode1 = 01 : fob
STPCSNU	EQU	18	subcode1 = 01 : char_set_num
STPPRIO	EQU	17	subcode1 = 01 : priority
STPPPOU	EQU	6	subcode1 = 01 : page printer
*			output
STPPRSA	EQU	7	subcode1 = 01 : print sample
STPTRAE	EQU	8	subcode1 = 01 : trace
STPACCE	EQU	9	subcode1 = 01 : access
STPREVI	EQU	20	subcode1 = 01 : revision
STPFORC	EQU	22	subcode1 = 01 : force
.Q01001	ANOP		
*			
STPFORCE	DS	FL1	type of use no
* _flag_set			
	AIF		('&EQUATES' NE 'YES').Q01002
STPFALL	EQU	0	_type = *all
STPFNON	EQU	1	_type = *none
STPFLOC	EQU	2	_type = *local
STPFREM	EQU	3	_type = *remote
STPFCEN	EQU	4	_type = *central
STPFDEV	EQU	64	_type = *device
STPFMDV	EQU	128	_type = *manag_dev
STPFWLD	EQU	5	_type = *wildcard
STPFLST	EQU	6	_type = *list
STPFSTD	EQU	7	_type = *std
STPFOWN	EQU	8	_type = *own
STPFHOM	EQU	9	_type = *home
STPFIDE	EQU	10	_type = *ident
STPFANY	EQU	11	_type = *any
STPFONL	EQU	12	_type = *only
STPFRNG	EQU	13	_type = *range
STPFONE	EQU	14	_type = *one
STPFNO	EQU	15	_type = *no
STPEQU	EQU	16	_type=*equivalent
STPEQUX	EQU	17	_type=*equivalent_ex
STPEXCE	EQU	18	_type=*except
STPALLO	EQU	19	_type=*allowed
STPFYES	EQU	20	_type = *yes
STPFCOM	EQU	21	_type = *complete
STPFSTA	EQU	22	_type = *status
STPFBLK	EQU	23	_type = *block_control
STPFMAN	EQU	24	_type = *manual
STPFM01	EQU	25	_type = *mode-1
STPFM02	EQU	26	_type = *mode2
STPFACT	EQU	27	_type = *active

```

STPFNOA EQU 28 _type = *no-active
STPFUNC EQU 29 _type = *unchanged
STPFADD EQU 30 _type = *add
STPFREV EQU 31 _type = *remove
STPFUPD EQU 32 _type = *update
STPFIMD EQU 33 _type = *immediate
.Q01002 ANOP
*
STPUN1 DS XL15
*
STPADVOUT DS 0XL132 device name
STPAFCB DS FL1
STPAUCB DS XL3
STPAECB DS 8CL8
          AIF ('&EQUATES' NE 'YES').Q01003
STPAECB# EQU 8
.Q01003 ANOP
STPAEC2 DS 8CL8
          AIF ('&EQUATES' NE 'YES').Q01004
STPAEC2# EQU 8
.Q01004 ANOP
*
          AIF ('&EQUATES' NE 'YES').Q01005
STP# EQU *-STPHDR
.Q01005 ANOP

```



## STRSPPR - Druckern Druckaufträge zuweisen

**Benutzergruppe:** Systembetreuer

**Programmiersprache:** Assembler, CPP, C

**Makrotyp:** M, S

Mit dem Makro STRSPPR werden Druckern Druckaufträge zugewiesen.

### Format

Operation	Operanden
STRSPPR	<p>ENTRY = <u>YES</u> / = NO</p> <p>VARIANT = &lt;c-string-without-quotes 3..3&gt; / <u>001</u></p> <p>DEVTYP = *ANY-LOCAL / *RSO / <u>*ANY-LOCAL</u></p> <p>DEVNAM = array(8): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>RSOPAR = &lt;var: pointer&gt; / <u>*NONE</u></p> <p>DESTTYP = <u>*NONE</u> / *STD / *LIST</p> <p>DESTNAM = array(16): &lt;var: char:8&gt; / *CENTRAL / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>FORMTYP = <u>*ALL</u> / *STD / *LIST / *EQUIVALENT / *EXCEPT / *EQUIVALENT_EX</p> <p>FORMNAM = array(16): &lt;var: char:6&gt; / &lt;c-string 1..6: c-string 1-6&gt;</p> <p>OVLTYTYP = <u>*ALL</u> / *NONE / *ONLY / *EXCEPT / *LIST</p> <p>OVLNAM = array(16): &lt;var: char:2&gt; / &lt;c-string 1..2: c-string 1-2&gt;</p> <p>HOSTTYP = <u>*ALL-CLUSTERS</u> / *HOME / *LOCAL-CLUSTER / *EXCEPT / *LIST / *CLIST</p>

(Teil 1 von 3)

Operation	Operanden
STRSPPR	<p>HOSTNAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>PAPROUT = <u>*NO</u> / *ALLOWED</p> <p>PSAMPLE = <u>*NO</u> / *YES</p> <p>TRACE = <u>*NO</u> / *COMPLETE / *STATUS / *BLOCK_CONTROL</p> <p>USERTYP = <u>*ALL</u> / *EXCEPT / *LIST / *CLIST</p> <p>USERNAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>CLASTYP = <u>*ALL</u> / *EXCEPT / *LIST</p> <p>CLASNAM = array(16): &lt;var: int:1&gt; / &lt;integer 1..255&gt;</p> <p>SPNATYP = <u>*ALL</u> / *EXCEPT / *LIST / *CLIST</p> <p>SPNANAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>ACCTYP = <u>*ALL</u> / *EXCEPT / *LIST</p> <p>ACCNAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>ROT = *NO / *YES / <u>*ANY</u> / *MANUAL</p> <p>TWO_UP = *NO / *YES / <u>*ANY</u> / *MODE-1 / *MODE-2</p> <p>F_O_B = (<i>type, low, high</i>):  <i>type</i> : <u>*ANY</u> / *ONLY / *NO / *RANGE  <i>low</i> : &lt;var: int:2&gt; / &lt;integer 0..32767&gt; / <u>*STD</u>  <i>high</i> : &lt;var: int:2&gt; / &lt;integer 0..32767&gt; / <u>*STD</u></p>

(Teil 2 von 3)

Operation	Operanden
STRSPPR	<p>PRIO = (type, low, high):                      type : <u>*ALL</u> / *RANGE                      from : &lt;var: int:1&gt; / &lt;integer 30..255&gt; / <u>*MIN</u>                      to : &lt;var: int:1&gt; / &lt;integer 30..255&gt; / <u>*MAX</u></p> <p>CHARSN = (type, low, high):                      type : <u>*ALL</u> / *ONE / *RANGE                      low : &lt;var: int:2&gt; / &lt;integer 1..32767&gt; / <u>*MIN</u>                      high : &lt;var: int:2&gt; / &lt;integer 1..32767&gt; / <u>*MAX</u></p> <p>EXITR = <u>*ACTIVE</u> / *NOT-ACTIVE</p>

(Teil 3 von 3)

### Operandenbeschreibung

#### ENTRY=YES/NO

Angabe zu korrespondierenden Einträgen

#### ENTRY=YES

Korrespondierende Einträge werden generiert.

#### ENTRY=NO

Korrespondierende Einträge werden nicht generiert.

#### VARIANT=001 / <c-string-without-quotes 3..3>

Bezeichnet die Variante der generierten Parameterliste.

#### DEVTYP=\*ANY-LOCAL / \*RSO

Legt den Gerätetyp fest.

#### DEVTYP=\*ANY-LOCAL

Es werden lokale Drucker zugewiesen.

#### DEVTYP=\*RSO

Es sollen RSO-Drucker zugewiesen werden.

#### DEVNAM=array(8): <alphanum-name 1..8> / <c-string 1..8: c-string 1..8>

Logischer Geräteiname, der in der SPOOL-Parameterdatei definiert ist (bis zu 8 alphanumerischen Zeichen).

Wenn ein Gerät mit einem bestimmten Namen aktiviert wird (START- oder MODIFY-PRINTER-OUTPUT), kann es nur unter diesem Namen adressiert werden bis es wieder deaktiviert wird (STOP-PRINTER-OUTPUT-Kommando oder STPSPPR-Makro).

**RSOPAR=\*NONE / <var:pointer>**

Adresse der RSO-Parameterliste. Sie wird nur angegeben, wenn bestimmte RSO-Parameter gebraucht werden.

**RSOPAR=\*NONE**

Es gibt keine Adresse für eine RSO-Parameterliste.

**RSOPAR=<var:pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die RSO-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die RSO-Parameterliste hinterlegt ist (A(feld)).

**DESTTYP=\*NONE / \*STD /\*LIST**

Legt den Typ des Ausgabe-Gerätepools fest.

**DESTTYP=\*NONE**

Es gibt keine Gerätepools, z. B. weil die Liste der lokalen Gerätepools für die spezifizierten Drucker gelöscht ist und darum SPOOLOUT-Aufträge für diese Pools nicht mehr durchgeführt werden.

**DESTTYP=\*STD**

Die Liste der lokalen Pools ist durch eine Liste der Standard-Pools ersetzt (SPOOL-Parameterdatei), die die spezifizierten Drucker enthalten.

**DESTTYP=\*LIST**

Liste der Pools, zu denen die SPOOLOUT-Aufträge für die spezifizierten Drucker gelenkt werden können.

**DESTNAM= array(16): <var:char:8> / \*CENTRAL / <c-string 1..8: c-string 1..8>**

Legt den Namen bzw. eine Liste von Namen für die Ausgabe-Gerätepools fest. Dorthin können SPOOLOUT-Aufträge der spezifizierten Drucker gelenkt werden .

**DESTNAM=\*CENTRAL**

Der Drucker ist fähig dazu, lokale SPOOLOUT-Aufträge ohne spezifische Angabe des Ausgabe-Gerätepools auszuführen.

**FORMTYP=\*ALL/\*STD/\*LIST/\*EQUIVALENT/\*EXCEPT/\*EQUIVALENT-EXCEPT**

Legt den Typ der Angabe für Formularnamen fest

**FORMTYP=\*ALL**

SPOOLOUT-Aufträge können durchgeführt werden, egal welches Formular vom spezifizierten Drucker angefordert wird.

**FORMTYP=\*STD**

SPOOLOUT-Aufträge können durchgeführt werden, und zwar mit allen Formularen, die in der SPOOL-Parameterdatei für die spezifizierten Drucker erlaubt sind.

**FORMTYP=\*LIST**

SPOOLOUT-Aufträge können durchgeführt werden, und zwar mit allen Formularen, die im FORMNAM-Operanden angegeben sind.

**FORMTYP=\*EQUIVALENT**

Äquivalente Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können. Die Druckausgabe wird bei Formularwechsel nicht unterbrochen.

**FORMTYP=\*EXCEPT**

Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden können

**FORMTYP=\*EQUIVALENT-EXCEPT**

Liste der äquivalenten Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden können

**FORMNAM=array(16): <var:char:6> / <c-string 1..6: c-string 1..6>**

Liste der Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**OVLTP=\*ALL/\*NONE/\*ONLY/\*EXCEPT/\*LIST**

Legt den Typ der Filmdias fest, mit denen die SPOOLOUT-Aufträge auf den spezifizierten Laserdruckern durchgeführt werden können.

**OVLTP=\*ALL**

Alle SPOOLOUT-Aufträge können mit jedem Filmdia auf den spezifizierten Druckern durchgeführt werden.

**OVLTP=\*NONE**

Nur SPOOLOUT-Aufträge, die keine Filmdias benutzen, können auf den spezifizierten Druckern durchgeführt werden.

**OVLTP=\*ONLY**

Nur SPOOLOUT-Aufträge, die Filmdias benutzen, können auf den spezifizierten Druckern durchgeführt werden.

**OVLTP=\*EXCEPT**

Name der Filmdias, mit denen auf den spezifizierten Druckern keine SPOOLOUT-Aufträge durchgeführt werden können

**OVLTP=\*LIST**

Liste von Filmdias, mit denen auf den spezifizierten Druckern SPOOLOUT-Aufträge durchgeführt werden können.

**OVLNAM=array(16): <var:char:2> / <c-string 1..2: c-string 1..2>**

Liste von Filmdias, mit denen auf den spezifizierten Druckern SPOOLOUT-Aufträge durchgeführt werden können.

**HOSTTYP=\*ALL-CLUSTERS / \*HOME / \*LOCAL-CLUSTER / \*EXCEPT / \*LIST / \*CLIST**

Legt fest, dass der Drucker alle SPOOLOUT-Aufträge verarbeiten kann, die von einem bestimmten Host kommen.

**HOSTTYP=\*ALL-CLUSTERS**

SPOOLOUT-Aufträge von allen Rechnern in allen Clustern werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*HOME**

Nur SPOOLOUT-Aufträge vom lokalen Rechner werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*LOCAL-CLUSTER**

SPOOLOUT-Aufträge von allen Rechnern im lokalen Cluster werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*EXCEPT**

SPOOLOUT-Aufträge von allen Rechnern - mit Ausnahme der in HOSTNAM angegebenen Liste - werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*LIST**

SPOOLOUT-Aufträge von der Rechnerliste im Operanden HOSTNAM.

**HOSTTYP=\*CLIST**

Nur SPOOLOUT-Aufträge, die von den Hosts in dieser Liste kommen, werden verarbeitet. Alle Elemente der Liste müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**HOSTNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

An dem angegebenen Drucker werden nur Druckaufträge von Hosts dieser Liste bearbeitet

**PAPROUT=\*NO / \*ALLOWED**

Legt fest, ob SPOOLOUT-Aufträge für Laserdrucker auch auf Zeilendrucker durchgeführt werden können.

**PSAMPLE=\*NO / \*YES**

Bestimmt, ob für die SPOOLOUT-Aufträge, die unter Berücksichtigung der Zuweisungs-Operanden zur Druckausgabe gelangen, ein Probedruck durchgeführt werden soll. Für Drucker des Typs LP65, für PCL-Drucker und für SPS-Drucker wird PRINT-SAMPLE=\*YES abgewiesen.

Für einen Probedruck auf HP- oder HP90-Drucker darf kein Zeichensatz mit Proportional-schrift verwendet werden. Die Zeichen der auszugebenden Datei werden in diesem Fall durch '\*' und '0' ersetzt und sind in Proportional-schrift unterschiedlich breit.

**TRACE=\*NO / COMPLETE / STATUS / \*BLOCK\_CONTROL**

Schaltet die Ablaufverfolgung ein oder aus. Name der Trace-Datei:

- für SPOOL:  
\$SYSSPOOL.SYSTRC.SPOOL.<dev-mnemonic>.<yyyy-mm-dd>.<hh-mm>
- für SPS:  
\$SYSSPOOL.SYSTRC.SPS.<dvcname>.<yyyy-mm-dd>.<hhmmss>.<W/S>  
(W für Winter/S für Sommer)
- für RSO:  
\$SYSPPOOL.SYSTRC.RSO.<dvcname>.<yyyy-mm-dd>.<ss.mm>

**TRACE=\*NO**

Der Ablauf soll nicht protokolliert werden.

**TRACE=COMPLETE**

Der Ablauf soll vollständig protokolliert werden.

**TRACE=STATUS**

Nur für SPS-Drucker: Ablauf- und Statusdaten sollen protokolliert werden.

**TRACE=\*BLOCK\_CONTROL**

Nur für SPS-Drucker: Ablauf-, Status- und Block-Kontrolldaten sollen protokolliert werden.

**USERTYP=\*ALL / \*EXCEPT / \*LIST / \*CLIST**

Typ der Benutzerkennungen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**USERTYP=\*ALL**

Unabhängig von der Benutzerkennung können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**USERTYP=\*EXCEPT**

Unter den angegebenen Benutzerkennungen können SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden.

**USERTYP=\*LIST**

Liste der Benutzerkennungen, die im Operanden USRNAM angegeben sind, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**USERTYP=\*CLIST**

Liste der Benutzerkennungs-Typen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können. Alle Elemente der Liste müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**USERNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Liste der Benutzerkennungen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*ALL/\*EXCEPT/\*LIST**

SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*ALL**

Unabhängig von der SPOOLOUT-Klasse können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**CLASTYP=\*EXCEPT**

SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden können.

**CLASTYP=\*LIST**

Liste der SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASNAM=array(16): <var:int:1>/<integer 1..255>**

Liste der SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*ALL / \*EXCEPT / \*LIST / \*CLIST**

Auftragsnamen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*ALL**

Unabhängig vom Auftragsnamen können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**SPNATYP=\*EXCEPT**

Unter den angegebenen Auftragsnamen können SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden.

**SPNATYP=\*LIST**

Liste der Auftragsnamen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*CLIST**

Liste von Auftragsnamen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können. Alle Elemente der Liste müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**SPNANAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Liste der Auftragsnamen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ACCTYP=\*ALL/\*EXCEPT/\*LIST**

Abrechnungsnummern der SPOOLOUT-Aufträge, die auf den angegebenen Druckern verarbeitet werden können.



**ACCTYP= \*ALL**

Unabhängig von der Abrechnungsnummer können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**ACCTYP=\*EXCEPT**

Mit den angegebenen Abrechnungsnummern können SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden.

**ACCTYP=\*LIST**

Liste der Abrechnungsnummern, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ACCNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Liste der Abrechnungsnummern, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern im verarbeitet werden können.

**ROT=\*ANY/\*NO/\*YES/\*MANUAL**

Bestimmt, ob auf den angegebenen Druckern SPOOLOUT-Aufträge, die das Seitendrehmodul benötigen, verarbeitet werden können.

**ROT=\*ANY**

Unabhängig davon, ob sie das Seitendrehmodul benötigen, können auf den angegebenen Druckern SPOOLOUT-Aufträge verarbeitet werden.

**ROT=\*NO**

Auf den angegebenen Druckern können nur SPOOLOUT-Aufträge verarbeitet werden, die das Seitendrehmodul nicht benötigen.

**ROT=\*YES**

Auf den angegebenen Druckern können nur SPOOLOUT-Aufträge verarbeitet werden, die das Seitendrehmodul benötigen.

**ROT=\*MANUAL**

Der Operator kann hier, im Unterschied zum Wert \*NO, manuell über Hardware-Schalter das Seitendrehmodul ansprechen. In diesem Fall werden alle Seiten gedreht ausgedruckt.

**TWO\_UP=\*ANY / \*YES / \*NO / \*MODE-1 / \*MODE-2**

Nur für lokalen SPOOL.

Legt für SPOOLOUT-Aufträge auf HP90-PRINTER die druckerspezifische Eigenschaft TWO-UP-PROCESSING fest:

Es wird festgelegt, ob und in welcher Abfolge zwei nebeneinander liegende Seiten ausgegeben werden sollen. Eine ausführliche Beschreibung der unterschiedlichen 'TWO-UP'-Modi finden Sie im Handbuch „[SPOOL \(BS2000/OSD\)](#)“.

**FOB=(type, low, high)**

Bestimmt, ob auf den angegebenen Druckern Druckaufträge, die ein FOB-Datendia verwenden, verarbeitet werden können.

**type: \*ANY/\*ONLY/\*NO/\*RANGE**

Art der Druckaufträge, die auf den angegebenen Druckern verarbeitet werden können.

**type: \*ANY**

Druckaufträge mit und ohne FOB-Datendia können auf den angegebenen Druckern verarbeitet werden. Der maximal mögliche FORMS-OVERLAY-BUFFER wird vom korrespondierenden Standard-Geräte-Eintrag entnommen.

**type: \*ONLY**

Nur Druckaufträge, die ein FOB-Datendia verwenden, können auf den angegebenen Druckern verarbeitet werden. Der maximal mögliche FORMS-OVERLAY-BUFFER wird vom korrespondierenden Standard-Geräte-Eintrag entnommen.

**type: \*NO**

Druckaufträge, die ein FOB-Datendia verwenden, können auf den angegebenen Druckern nicht verarbeitet werden.

**type: \*RANGE**

Nur Druckaufträge, die ein FOB-Datendia verwenden, dessen Größe innerhalb des angegebenen Intervalls liegt, können auf den angegebenen Druckern verarbeitet werden.

**low: \*STD/ <var:int:2> / <integer 0..32767>/**

Untergrenze des Intervalls (Angabe in Unterzeilen). \*STD entspricht dabei dem Wert 0.

**high: \*STD / <var:int:2> / <integer 0..32767>**

Obergrenze des Intervalls für HP90-Drucker: 4032 Unterzeilen. Obergrenze des Intervalls für HP-PRINTER: 672 Unterzeilen. \*STD entspricht dabei dem Wert 4032.

**PRIO=(type, low, high)**

Prioritäten der Druckaufträge, die auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL/\*RANGE**

Art der Druckaufträge, die auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL**

Druckaufträge können unabhängig von ihrer Priorität auf den angegebenen Druckern verarbeitet werden.

**type: \*RANGE**

Druckaufträge, deren Priorität innerhalb des angegebenen Intervalls liegt, können auf den angegebenen Druckern verarbeitet werden.

**low: \*MIN/ <var: int: 1>/ <integer 30..255>**

Untergrenze des Intervalls. \*MIN entspricht dabei dem Wert 30.

**high: \*MAX** / <var:int:1> / <integer 30..255>

Obergrenze des Intervalls. \*MAX entspricht dem Wert 255.

**CHARSN=(type, low, high)**

Anzahl der benötigten Zeichensätze, mit denen Druckaufträge auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL** / \*ONE / \*RANGE

Zeichensätze, mit denen Druckaufträge auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL**

Unabhängig von der Anzahl der Zeichensätze können Druckaufträge auf den angegebenen Druckern verarbeitet werden. Die maximal mögliche Anzahl der benötigten Zeichensätze wird vom korrespondierenden Standard-Geräte-Eintrag entnommen.

**type: \*ONE**

Nur Druckaufträge, die nicht mehr als einen Zeichensatz benötigen, können auf den angegebenen Druckern verarbeitet werden.

**type: \*RANGE**

Druckaufträge, deren Anzahl benötigter Zeichensätze innerhalb des angegebenen Intervalls liegt, können auf den angegebenen Druckern verarbeitet werden. Während der Initialisierung eines Geräts wird geprüft, wieviele Zeichensätze auf dem Gerät geladen werden können.

Ist dieser Wert kleiner als einer der beiden angegebenen Werte, wird entweder eine Meldung an die Bedienstation geschickt und der Wert für *high* angepasst (wenn  $low \leq$  Anzahl der Zeichensätze, die auf dem Gerät geladen werden können), oder das Kommando wird abgewiesen (wenn  $low >$  Anzahl der Zeichensätze, die auf dem Gerät geladen werden können).

**low: \*MIN** / <var:int:2> / <integer 1..32767>

Untergrenze des Intervalls. \*MIN entspricht dabei dem Wert 1.

**high: \*MAX** / <var:int:2> / <integer 1..32767>

Obergrenze des Intervalls. \*MAX entspricht dabei dem Wert 64.

**EXITR=\*ACTIVE** / \*NOT-ACTIVE

Gibt an, ob EXIT-Routinen beim SPOOLOUT aufgerufen werden sollen.

## Returncodes

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	Kein Fehler
02	00	xxxx	Warnung - Drucker existiert nicht oder Ausführung ist nicht erlaubt
02	01	FFFF	Ungültige Parameterlisteara
01	01	xxxx	Ungültiger DEVICE-Operand
02	01	xxxx	Ungültiger DESTINATION-Operand
03	01	xxxx	Ungültiger FORM-NAME-Operand
04	01	xxxx	Ungültiger OVERLAY-Operand
05	01	xxxx	Ungültiger HOST-NAME-Operand
06	01	xxxx	Ungültiger PAGE-PRINTER-OUTPUT-Operand
07	01	xxxx	Ungültiger PRINT-SAMPLE-Operand
08	01	xxxx	Ungültiger TRACE-Operand
09	01	xxxx	Ungültiger ACCESS-Operand
0A	01	xxxx	Ungültiger USERID-Operand
0B	01	xxxx	Ungültiger SPOOLOUT-CLASS-Operand
0C	01	xxxx	Ungültiger JOB-NAME-Operand
0D	01	xxxx	Ungültiger ACCOUNT-Operand
0E	01	xxxx	Ungültiger ROTATION-Operand
0F	01	xxxx	Ungültiger TWO-UP -Operand
10	01	xxxx	Ungültiger FORM-OVERLAY-BUFFER-Operand
11	01	xxxx	Ungültiger PRIORITY-Operand
12	01	xxxx	Ungültiger CHARACTER-SET-NUMBER-Operand
13	01	xxxx	Ungültige Adresse der RSO-Parameterlistenerweiterung
14	01	xxxx	Ungültiger REVISION-Operand
15	01	xxxx	Ungültiger EXIT-Operand
17	01	xxxx	Ungültige Adresse von dprintpl
18	01	xxxx	Ungültige Adresse von sps pl
1A	01	xxxx	Ungültige Adresse von dsem pl
1B	01	xxxx	Ungültige Pufferadresse
1C	01	xxxx	Ungültige Pufferlänge
1D	01	xxxx	Ungültiger SCHEDULING-STATE-Operand
1E	01	xxxx	Ungültiger STATUS-Operand
1F	01	xxxx	Ungültiger CLUSTER-Operand
20	01	xxxx	Ungültiger SERVER-Operand
21	01	xxxx	Ungültiger ERROR-MESSAGE-Operand
00	20	xxxx	Systemfehler
02	40	xxxx	Keine Berechtigung
04	40	xxxx	Operandenwertfehler
05	40	xxxx	Parameterlistenversion wird nicht unterstützt
02	80	xxxx	RSO-Subsystem ist nicht geladen
05	80	xxxx	SPS-Subsystem ist nicht geladen
06	80	xxxx	Dprintcl-Subsystem ist nicht geladen
07	80	xxxx	Dprintcm-Subsystem ist nicht geladen
08	80	xxxx	Dprintsv-Subsystem ist nicht geladen
FF	FF	FFFF	Spool-Subsystem ist nicht geladen

*Hinweis*

Die vier Zeichen xxxx in der Maincode-Spalte bezeichnen eine Meldungskennung. Um eine ausführlichere Information über einen Fehler zu erhalten, geben Sie das Kommando HELP-MSG SCPxxxx oder SPSxxxx im SYSTEM MODUS ein.

**Beschreibung der Parameterliste**

```

*   _dev_type_set
      AIF ('&EQUATES' NE 'YES').Q01001
STRANYL   EQU   1                      local device
STRRSO    EQU   2                      RSO device
.Q01001  ANOP
*
*   parameter list description
STR_MDL   DS    0F
STRHDR    FHDR MF=(C,STR),EQUATES=NO    Standard header
*   main return codes
      AIF ('&EQUATES' NE 'YES').Q01002
STRRCNO   EQU   0                      subcode1 = 00 : no error
STRWARN   EQU   2                      printer not found or
*                                               processing not permitted
STRVALM   EQU   2                      subcode1 = 01 : pl area
*                                               invalid - maincode = FFFF
STRPARM   EQU   1                      subcode1 = 01 : wrong
*                                               parameter
STRDEVI   EQU   1                      subcode1 = 01 : device
STRDEST   EQU   2                      subcode1 = 01 : destination
STRFONA   EQU   3                      subcode1 = 01 : form name
STROVRL   EQU   4                      subcode1 = 01 : overlay
STRHOST   EQU   5                      subcode1 = 01 : host
STRPPOU   EQU   6                      subcode1 = 01 : page printer
*                                               output
STRPRSA   EQU   7                      subcode1 = 01 : print sample
STRTRAE   EQU   8                      subcode1 = 01 : trace
STRACCE   EQU   9                      subcode1 = 01 : access
STRUSER   EQU  10                      subcode1 = 01 : userid
STRSPCL   EQU  11                      subcode1 = 01 : spoolout
*                                               class
STRJNAM   EQU  12                      subcode1 = 01 : job name
STRACCN   EQU  13                      subcode1 = 01 : account
STRROTA   EQU  14                      subcode1 = 01 : rotation
STRTWOU   EQU  15                      subcode1 = 01 : two-up
STRFOB    EQU  16                      subcode1 = 01 : fob
STRPRIO   EQU  17                      subcode1 = 01 : priority
STRCSNU   EQU  18                      subcode1 = 01 : char_set_num
STRRSOL   EQU  19                      subcode1 = 01 : addr of rso

```

```

*
STRREVI    EQU    20
STRXTT     EQU    21
STRSYSE    EQU    0
STRPRIV    EQU    2
*
STRVERR    EQU    4
STRVERE    EQU    5
STRNOSS    EQU    0
*
STRNRSO    EQU    2
STRNDCL    EQU    6
STRNDCM    EQU    7
STRNDSV    EQU    8
STRNSPO    EQU    255
*
.Q01002    ANOP
*
*
STRADVNM   DS     0XL132
STRAF8B    DS     FL1
*   _flag_set
        AIF ('&EQUATES' NE 'YES').Q01003
STRFALL    EQU    0
STRFNON    EQU    1
STRFLOC    EQU    2
STRFREM    EQU    3
STRFCEN    EQU    4
STRFDEV    EQU    64
STRFMDV    EQU    128
STRFWLD    EQU    5
STRFLST    EQU    6
STRFSTD    EQU    7
STRFOWN    EQU    8
STRFHOM    EQU    9
STRFIDE    EQU    10
STRFANY    EQU    11
STRFONL    EQU    12
STRFRNG    EQU    13
STRFONE    EQU    14
STRFNO     EQU    15
STREQU     EQU    16
STREQUX    EQU    17
STREXCE    EQU    18
STRALLO    EQU    19
STRFYES    EQU    20
STRFCOM    EQU    21
STRFSTA    EQU    22

pl
subcode1 = 01 : revision
subcode1 = 01 : exit
subcode1 = 20 : system error
subcode1 = 40 : no
authorization
operand value error
p/l version not supported
subcode1 = 80 : subsystem not
loaded
rso not loaded
dprintcl not loaded
dprintcm not loaded
dprintsv not loaded
subcode1 = 255: spool not
loaded
_type = *all
_type = *none
_type = *local
_type = *remote
_type = *central
_type = *device
_type = *manag_dev
_type = *wildcard
_type = *list
_type = *std
_type = *own
_type = *home
_type = *ident
_type = *any
_type = *only
_type = *range
_type = *one
_type = *no
_type=*equivalent
_type=*equivalent_ex
_type=*except
_type=*allowed
_type = *yes
_type = *complete
_type = *status

```

```

STRFBLK    EQU    23                _type = *block_control
STRFMAN    EQU    24                _type = *manual
STRFM01    EQU    25                _type = *mode-1
STRFM02    EQU    26                _type = *mode2
STRFACT    EQU    27                _type = *active
STRFNOA    EQU    28                _type = *no-active
STRFUNC    EQU    29                _type = *unchanged
STRFADD    EQU    30                _type = *add
STRFREV    EQU    31                _type = *remove
STRFUPD    EQU    32                _type = *update
STRFMPR    EQU    30                _type = *minprio
STRFMAP    EQU    255              _type = *maxprio
STRFMCH    EQU    1                 _type = *minchars
STRFMXC    EQU    64                _type = *maxchars
.Q01003    ANOP
*
STRAUC8B   DS     XL3
STRAEC8B   DS     8CL8
           AIF    ('&EQUATES' NE 'YES').Q01004
STRAEC8B#  EQU    8
.Q01004    ANOP
STRAEC82   DS     8CL8
           AIF    ('&EQUATES' NE 'YES').Q01005
STRAEC82#  EQU    8
.Q01005    ANOP
*
*
STRDDEST   DS     0XL132            destination
STRDFC8    DS     FL1
STRDUC8    DS     XL3
STRDEC8    DS     16CL8
           AIF    ('&EQUATES' NE 'YES').Q01006
STRDEC8#   EQU    16
.Q01006    ANOP
*
*
STRAFRMN   DS     0XL100            form name
STRAFC6    DS     FL1
STRAUC6    DS     XL3
STRAEC6    DS     16CL6
           AIF    ('&EQUATES' NE 'YES').Q01007
STRAEC6#   EQU    16
.Q01007    ANOP
*
*
STRAFOVL   DS     0XL36            form overlay
STRAFC2    DS     FL1
STRAUC2    DS     XL3

```

```

STRAEC2   DS    16CL2
           AIF   ('&EQUATES' NE 'YES').Q01008
STRAEC2#  EQU   16
.Q01008   ANOP
*
*
STRBHOST  DS    0XL132                host name
STRBFC8   DS    FL1
STRBUC8   DS    XL3
STRBEC8   DS    16CL8
           AIF   ('&EQUATES' NE 'YES').Q01009
STRBEC8#  EQU   16
.Q01009   ANOP
*
*
STRPPO    DS    FL1                page printer output
STRPSAM   DS    FL1                print sample
STRTRAC   DS    FL1                trace
STRUN1    DS    XL1
*
*
STREACCS  DS    0XL20                allowed accesses
STREFC4   DS    FL1
STREUC4   DS    XL3
STREEC4   DS    4CL4
           AIF   ('&EQUATES' NE 'YES').Q01010
STREEC4#  EQU   4
.Q01010   ANOP
*
*
*
STRCUSRD  DS    0XL132                userid
STRCFC8   DS    FL1
STRCUC8   DS    XL3
STRCEC8   DS    16CL8
           AIF   ('&EQUATES' NE 'YES').Q01011
STRCEC8#  EQU   16
.Q01011   ANOP
*
*
*
STRASPCL  DS    0XL20                spoolout class
STRAFI1   DS    FL1
STRAUC1   DS    XL3
STRAEI1   DS    16X
*
*
*
STRESPNA  DS    0XL132                spoolout name
STREFC8   DS    FL1
STREUC8   DS    XL3
STREEC8   DS    16CL8
           AIF   ('&EQUATES' NE 'YES').Q01012

```



```

STREEC8# EQU 16
.Q01012 ANOP
*
*
STRFACNT DS 0XL132 account
STRFFC8 DS FL1
STRFUC8 DS XL3
STRFEC8 DS 16CL8
AIF ('&EQUATES' NE 'YES').Q01013
STRFEC8# EQU 16
.Q01013 ANOP
*
STRROT DS FL1 rotation
STRTUP DS FL1 two-up processing
STRUN2 DS XL2
*
STRMFOB DS 0XL8 form-overlay-buffer
STRMFR2 DS FL1
STRMUN1 DS XL1
STRMLR2 DS H
STRMHR2 DS H
STRMUR2 DS XL2
*
*
STROPRIO DS 0XL4 priority
STROFR1 DS FL1
STROLR1 DS X
STROHR1 DS X
STROUR1 DS XL1
*
*
STRACHAR DS 0XL8 character-set-number
STRAFR2 DS FL1
STRAUN1 DS XL1
STRALR2 DS H
STRAHR2 DS H
STRAUR2 DS XL2
*
STRRSOP DS A address of rso_p1
STRPTRRES1 DS A address of reserved1
STRPTRRES2 DS A address of reserved2
STRPTRRES3 DS A address of reserved3
STRREVIS DS X revision number
STRUN4 DS XL3
STREXIT DS FL1 exit-processing
STRSP1 DS AL1 specified1
AIF ('&EQUATES' NE 'YES').Q01014
STRSP11 EQU X'80' device type

```

```

STRSP12    EQU    X'40'                rso param
STRSP13    EQU    X'20'                destination type
STRSP14    EQU    X'10'                form type
STRSP15    EQU    X'08'                overlay type
STRSP16    EQU    X'07'                unused
.Q01014    ANOP
STRSP2     DS     AL1                   specified2
          AIF    ('&EQUATES' NE 'YES').Q01015
STRSP21    EQU    X'80'                host type
STRSP22    EQU    X'40'                page pr. output
STRSP23    EQU    X'20'                print sample
STRSP24    EQU    X'10'                trace
STRSP25    EQU    X'08'                userid type
STRSP26    EQU    X'07'                unused
.Q01015    ANOP
STRSP3     DS     AL1                   specified3
          AIF    ('&EQUATES' NE 'YES').Q01016
STRSP31    EQU    X'80'                spoolout class type
STRSP32    EQU    X'40'                spoolout name type
STRSP33    EQU    X'20'                spoolout name list
STRSP34    EQU    X'10'                account type
STRSP35    EQU    X'08'                rotation
STRSP36    EQU    X'04'                two-up processing
STRSP37    EQU    X'02'                charsetnum low
STRSP38    EQU    X'01'                charsetnum high
.Q01016    ANOP
STRSP4     DS     AL1                   specified4
          AIF    ('&EQUATES' NE 'YES').Q01017
STRSP41    EQU    X'80'                fob type
STRSP42    EQU    X'40'                fob range low
STRSP43    EQU    X'20'                fob range high
STRSP44    EQU    X'10'                priority
STRSP45    EQU    X'08'                priority from
STRSP46    EQU    X'04'                priority to
STRSP47    EQU    X'02'                charsetnum
STRSP48    EQU    X'01'                exit routines
.Q01017    ANOP
STRUN3     DS     XL43
          AIF    ('&EQUATES' NE 'YES').Q01018
STR#       EQU    *-STRHDR
.Q01018    ANOP

```

## STRSRSO - Drucker mit RSO-spezifischen Parametern zuweisen

**Benutzergruppe:** Systembetreuer, RSO-Geräteverwalter

**Programmiersprache:** Assembler, CPP, C

**Makrotyp:** M, S

Mit dem Makro STRSRSO werden Drucker RSO-Druckaufträge zugewiesen.

### Format

Operation	Operanden
STRSRSO	VARIANT = <u>001</u> / <c-string-without-quotes 3..3>  ACSTYP = <u>*STD</u> / *LIST  ACSNAM = array(4): <var: char:4> / <c-string 1..8: c-string 1-4>

### Operandenbeschreibung

**VARIANT=001 / <c-string-without-quotes 3..3>**

Bezeichnet die Variante der generierten Parameterliste.

**ACSTYP=\*STD / \*LIST**

Dieser Operand legt fest, welche Zugriffe erlaubt sind. Er ist für jeden RSO-Drucker zulässig

**ACSTYP=\*STD**

Nur RSO-Aufträge dürfen auf dem Drucker durchgeführt werden.

**ACSTYP=\*LIST**

Liste der erlaubten Zugriffe, die im Operanden ACSNAM angegeben ist.

**ACSNAM=array(4): <var:char:4> / <c-string 1..8: c-string 1..4>**

Name des erlaubten Zugriffs. Es sind folgende Angaben möglich:

C`RSO` RSO-Druckaufträge können dem gestarteten Drucker zugewiesen werden.

C`UTM` UTM-Druckaufträge können dem gestarteten Drucker zugewiesen werden

**Beschreibung der Parameterliste**

```

*   parameter list description
SSR_MDL DS    OF
SSRHDR  FHDR MF=(C,SSR),EQUATES=NO    Standard header
*   main return codes
      AIF  ('&EQUATES' NE 'YES').Q01001
SSRRCNO EQU  0                          subcode1 = 00 : no error
SSRWARN EQU  2                          job not found or processing
*                                           not permitted
SSRPARM EQU  1                          subcode1 = 01 : wrong
*                                           parameter
SSRVALD EQU  2                          invalid address/length
SSRSHRT EQU  3                          output area too short
SSRSYSE EQU  0                          subcode1 = 20 : system error
SSRPRIV EQU  2                          subcode1 = 40 : no
*                                           authorization
SSRVERR EQU  4                          operand value error
SSRVERE EQU  5                          p/l version not supported
SSRNOSS EQU  0                          subcode1 = 80 : subsystem not
*                                           loaded
SSRNspo EQU  1                          spool not loaded
SSRNRSO EQU  2                          rso not loaded
SSRNPLM EQU  3                          plam not loaded
SSRNJV  EQU  4                          jv not loaded
SSRNspS EQU  5                          sps not loaded
SSRNDCL EQU  6                          dprintc1 not loaded
SSRNDcm EQU  7                          dprintcm not loaded
SSRNDSV EQU  8                          dprintsv not loaded
SSRNDSM EQU  9                          dsem not loaded
.Q01001 ANOP
*
*
SSRAACCS DS    0XL20                      allowed access
SSRAFC4  DS    FL1
*   _flag_set
      AIF  ('&EQUATES' NE 'YES').Q01002
SSRFALL EQU  0                          _type = *all
SSRFNON EQU  1                          _type = *none
SSRFLOC EQU  2                          _type = *local
SSRFREM EQU  3                          _type = *remote
SSRFCEN EQU  4                          _type = *central
SSRFDEV EQU  64                          _type = *device
SSRFMDV EQU 128                          _type = *manag_dev
SSRFWLD EQU  5                          _type = *wildcard
SSRFLST EQU  6                          _type = *list
SSRFSTD EQU  7                          _type = *std
SSRFOWN EQU  8                          _type = *own

```

```

SSRFHOM EQU 9           _type = *home
SSRFIDE EQU 10          _type = *ident
SSRFANY EQU 11          _type = *any
SSRFONL EQU 12          _type = *only
SSRFRNG EQU 13          _type = *range
SSRFONE EQU 14          _type = *one
SSRFNO EQU 15           _type = *no
SSREQU EQU 16           _type=*equivalent
SSREQUX EQU 17          _type=*equivalent_ex
SSREXCE EQU 18          _type=*except
SSRALLO EQU 19          _type=*allowed
SSRFYES EQU 20          _type = *yes
SSRFCOM EQU 21          _type = *complete
SSRFSTA EQU 22          _type = *status
SSRFBLK EQU 23          _type = *block_control
SSRFMAN EQU 24          _type = *manual
SSRFM01 EQU 25          _type = *mode-1
SSRFM02 EQU 26          _type = *mode2
SSRFACT EQU 27          _type = *active
SSRFNOA EQU 28          _type = *no-active
.Q01002 ANOP
*
SSRAUC4 DS XL3
SSRAEC4 DS 4CL4
        AIF ('&EQUATES' NE 'YES').Q01003
SSRAEC4# EQU 4
.Q01003 ANOP
*
SSRUN1 DS XL40
        AIF ('&EQUATES' NE 'YES').Q01004
SSR# EQU *-SSRHDR
.Q01004 ANOP

```

## SUPSPPR - Verwaltungsparameter für Drucker ändern

**Benutzergruppe:** Systembetreuer, RSO-Geräteverwalter

**Programmiersprache:** Assembler, CPP, C

**Makrotyp:** M, S

Mit dem Makro SUPSPPR kann bei einem Drucker eine Änderung vorgenommen werden, der vorher mit STRSPPR(X) oder mit dem Kommando START-PRINTER-OUTPUT gestartet wurde.

### Format

Operation	Operanden
SUPSPPR	<p>ENTRY = <u>YES</u> / = NO</p> <p>VARIANT = &lt;c-string-without-quotes 3..3&gt; / <u>001</u></p> <p>DEVTYP = *ANY-LOCAL / *RSO / <u>UNCHANGED</u></p> <p>RSOPAR = &lt;var: pointer&gt; / *NONE / <u>UNCHANGED</u></p> <p>DEVNAM = array(8): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>DESTTYP = *NONE / *STD / *ADD / *REMOVE / *LIST / <u>UNCHANGED</u></p> <p>DESTNAME = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>FORMTYP = *ALL / *STD / *LIST / *EQUIVALENT / *EXCEPT / *EQUIVALENT_EX / *ADD / *REMOVE / <u>UNCHANGED</u></p> <p>FORMNAM = array(16): &lt;var: char:6&gt; / &lt;c-string 1..6: c-string 1-6&gt;</p> <p>OVLTYTYP = *ALL / *NONE / *ONLY / *EXCEPT / *LIST / *ADD / *REMOVE / <u>UNCHANGED</u></p> <p>OVLNAM = array(16): &lt;var: char:2&gt; / &lt;c-string 1..2: c-string 1-2&gt;</p>

(Teil 1 von 3)

Operation	Operanden
SUPSPPR	<p>HOSTTYP = *ALL-CLUSTER / *HOME / *LOCAL-CLUSTER  /*EXCEPT /*LIST /*ADD /*REMOVE / <u>*UNCHANGED</u> /*CLIST</p> <p>HOSTNAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>PSAMPLE = *NO / *YES / <u>*UNCHANGED</u></p> <p>TRACE = *NO / *COMPLETE / *STATUS / *BLOCK_CONTROL /  <u>*UNCHANGED</u></p> <p>USERTYP = *ALL / *EXCEPT /*LIST /*ADD /*REMOVE /  <u>*UNCHANGED</u> /*CLIST</p> <p>USERNAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>CLASTYP = *ALL / *EXCEPT /*LIST /*ADD /*REMOVE /  <u>*UNCHANGED</u></p> <p>CLASNAM = array(16): &lt;var: int:1&gt; / &lt;integer 1..255&gt;</p> <p>SPNATYP = *ALL / *EXCEPT /*LIST /*ADD /*REMOVE /  <u>*UNCHANGED</u> /*CLIST</p> <p>SPNANAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>ACCTYP = *ALL / *EXCEPT /*LIST /*ADD /*REMOVE /  <u>*UNCHANGED</u></p> <p>ACCNAM = array(16): &lt;var: char:8&gt; / &lt;c-string 1..8: c-string 1-8&gt;</p> <p>ROT = *NO / *YES / *ANY / *MANUAL / <u>*UNCHANGED</u></p> <p>TWO_UP = *NO / *YES / *ANY / *MODE-1 / *MODE-2 / <u>*UNCHANGED</u></p>

(Teil 2 von 3)

Operation	Operanden
SUPSPPR	<p>F_O_B = (type, low, high)  type: *ANY / *ONLY / *NO / *RANGE / <u>*UNCHANGED</u>  low: *STD / &lt;var: int:2&gt; / &lt;integer 0..32767&gt; / <u>*UNCHANGED</u>  high: *STD / &lt;var: int:2&gt; / &lt;integer 0..32767&gt; / <u>*UNCHANGED</u></p> <p>PRIO = (type, from, to)  type: *ALL / *RANGE / <u>*UNCHANGED</u>  from: *MIN / &lt;var: int:1&gt; / &lt;integer 30..255&gt; / <u>*UNCHANGED</u>  to: *MAX / &lt;var: int:1&gt; / &lt;integer 30..255&gt; / <u>*UNCHANGED</u></p> <p>CHARSN = (type, low, high):  type: *ALL / *ONE / *RANGE / <u>*UNCHANGED</u>  low: &lt;var: int:2&gt; / &lt;integer 1..32767&gt; / *MIN / <u>*UNCHANGED</u>  high: &lt;var: int:2&gt; / &lt;integer 1..32767&gt; / *MAX / <u>*UNCHANGED</u></p> <p>EXITR = *ACTIVE / *NOT-ACTIVE / <u>*UNCHANGED</u></p> <p>RVSION = &lt;var: int:1&gt; / &lt;integer 1..255&gt;</p>

(Teil 3 von 3)

## Operandenbeschreibung

### ENTRY=YES/NO

Angabe zu korrespondierenden Einträgen

### ENTRY=YES

Korrespondierende Einträge werden generiert.

### ENTRY=NO

Korrespondierende Einträge werden nicht generiert.

### VARIANT=001 / <c-string-without-quotes 3..3>

Bezeichnet die Variante der generierten Parameterliste.

### DEVTYP=\*ANY-LOCAL / \*RSO / \*UNCHANGED

Legt den Gerätetyp fest.

### DEVTYP=\*ANY-LOCAL

Für lokale Drucker, die im weiteren genauer bestimmt werden, sollen Verwaltungsparameter geändert werden.

### DEVTYP= \*RSO

Für RSO-Drucker, die im weiteren genauer bestimmt werden, sollen Verwaltungsparameter geändert werden.



**DEV TYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**DEVNAM=array(8): <alphanum-name 1..8> / <c-string 1..8: c-string 1..8>**

Logischer Gerätenamen, der in der SPOOL-Parameterdatei definiert ist (bis zu 8 alphanumerischen Zeichen).

Wenn ein Gerät mit einem bestimmten Namen aktiviert wird (START- oder MODIFY-PRINTER-OUTPUT), kann es nur unter diesem Namen adressiert werden, bis es wieder deaktiviert wird (STOP-PRINTER-OUTPUT-Kommando oder STPSPPR-Makro).

**RSOPAR=\*NONE / <var:pointer> / \*UNCHANGED**

Adresse der RSO-Parameterliste. Sie wird nur angegeben, wenn bestimmte RSO-Parameter gebraucht werden.

**RSOPAR=\*NONE**

Es gibt keine Adresse für eine RSO-Parameterliste.

**RSOPAR=<var:pointer>**

Es wird ein Zeiger vereinbart, d.h. der Inhalt der Variablen oder des Feldes ist nicht die RSO-Parameterliste selbst, sondern die Adresse eines Speicherplatzes, an dem die RSO-Parameterliste hinterlegt ist (A(feld)).

**RSOPAR=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert

**DESTTYP=\*NONE / \*STD / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED**

Legt den Typ des Ausgabe-Gerätepools fest.

**DESTTYP=\*NONE**

Es gibt keine Gerätepools, z. B. weil die Liste der lokalen Gerätepools für die spezifizierten Drucker gelöscht ist und darum Druckaufträge für diese Pools nicht mehr durchgeführt werden.

**DESTTYP=\*STD**

Die Liste der lokalen Pools ist durch eine Liste der Standard-Pools ersetzt (SPOOL-Parameterdatei), die die spezifizierten Drucker enthalten.

**DESTTYP=\*LIST**

Liste der Pools, zu denen die Druckaufträge für die spezifizierten Drucker gelenkt werden können.

**DESTTYP=\*ADD**

Pools, die zu einer bestehenden Liste von Pools hinzugefügt werden sollen, auf die Druckaufträge für die angegebenen Drucker gelenkt werden können.

**DESTTYP=\*REMOVE**

Pools, die aus einer bestehenden Liste von Pools gelöscht werden sollen, auf die Druckaufträge für die angegebenen Drucker gelenkt werden können.

**DESTTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**DESTNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Legt den Namen bzw. eine Liste von Namen für die Ausgabe-Gerätepools fest. Dorthin können SPOOLOUT-Aufträge der spezifizierten Drucker gelenkt werden.

**FORMTYP=\*ALL / \*STD / \*LIST / \*EQUIVALENT / \*EXCEPT / \*EQUIVALENT-EXCEPT / \*ADD / \*REMOVE / \*UNCHANGED**

Legt den Typ des Formulars in Verbindung mit dem Druckauftrag fest.

**FORMTYP=\*ALL**

SPOOLOUT-Aufträge können durchgeführt werden, egal welches Formular vom spezifizierten Drucker angefordert wird.

**FORMTYP=\*STD**

SPOOLOUT-Aufträge können durchgeführt werden, und zwar mit allen Formularen, die in der SPOOL-Parameterdatei für die spezifizierten Drucker erlaubt sind.

**FORMTYP=\*LIST**

SPOOLOUT-Aufträge können durchgeführt werden, und zwar mit allen Formularen, die in FORMTYP angegeben sind.

**FORMTYP=\*EQUIVALENT**

Mit äquivalenten Formularen können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden. Die Druckausgabe wird bei Formularwechsel nicht unterbrochen.

**FORMTYP=\*EXCEPT**

Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden können.

**FORMTYP=\*EQUIVALENT-EXCEPT**

Äquivalente Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden können.

**FORMTYP=\*ADD**

Formulare, die zu einer bestehenden Liste von Formularen hinzugefügt werden sollen, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**FORMTYP=\*REMOVE**

Formulare, die aus einer bestehenden Liste von Formularen gelöscht werden sollen, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**FORMTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht veraendert.

**FORMNAM=array(16): <var:char:6> / <c-string 1..6: c-string 1..6>**

Liste der Formulare, mit denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können. Existiert bereits eine (positive) Liste von Formularen für die angegebenen Drucker, wird sie durch diese neue Liste ersetzt. Die Druckausgabe wird bei jedem Formularwechsel mit einer Meldung an den Bedienplatz unterbrochen.

**OVLTYP=\*ALL / \*NONE / \*ONLY / \*EXCEPT / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED**

Legt den Typ der Filmdias fest, mit denen die SPOOLOUT-Aufträge auf den spezifizierten Laserdruckern durchgeführt werden können.

**OVLTYP=\*ALL**

Alle SPOOLOUT-Aufträge können mit jedem Filmdia auf den spezifizierten Druckern durchgeführt werden.

**OVLTYP=\*NONE**

Nur SPOOLOUT-Aufträge, die keine Filmdias benutzen, können auf den spezifizierten Druckern durchgeführt werden.

**OVLTYP=\*ONLY**

Nur SPOOLOUT-Aufträge, die Filmdias benutzen, können auf den spezifizierten Druckern durchgeführt werden.

**OVLTYP=\*EXCEPT**

Name der Filmdias, mit denen auf den spezifizierten Druckern keine SPOOLOUT-Aufträge durchgeführt werden können

**OVLTYP=\*LIST**

Liste von Filmdias, mit denen auf den spezifizierten Druckern SPOOLOUT-Aufträge durchgeführt werden können.

**OVLTYP=\*ADD**

Filmdias, die zu einer bestehenden Liste von Filmdias hinzugefügt werden sollen, mit denen SPOOLOUT-Aufträge auf den angegebenen Laser-Druckern verarbeitet werden können.

**OVLTYP=\*REMOVE**

Filmdias, die aus einer bestehenden Liste von Filmdias gelöscht werden sollen, mit denen SPOOLOUT-Aufträge auf den angegebenen Laser-Druckern verarbeitet werden können.

**OVLTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**OVLNAM=array(16): <var:char:2> / <c-string 1..2: c-string 1..2>**

Liste von Filmdias, mit denen auf den spezifizierten Druckern SPOOLOUT-Aufträge durchgeführt werden können.

**HOSTYP=\*ALL-CLUSTERS / \*HOME / \*LOCAL-CLUSTER / \*EXCEPT / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED / \*CLIST**

Legt fest, dass der Drucker alle SPOOLOUT-Aufträge ausführen kann, die von einem bestimmten Host aus gestartet werden.

**HOSTTYP=\*ALL-CLUSTERS**

SPOOLOUT-Aufträge von allen Rechnern in allen Clustern werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*HOME**

Nur SPOOLOUT-Aufträge vom lokalen Rechner werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*LOCAL-CLUSTER**

SPOOLOUT-Aufträge von allen Rechnern im lokalen Cluster werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*EXCEPT**

SPOOLOUT-Aufträge von allen Rechnern mit Ausnahme der in HOSTNAM angegebenen Liste werden auf dem spezifizierten Drucker durchgeführt.

**HOSTTYP=\*LIST**

SPOOLOUT-Aufträge von der Rechnerliste im Operanden HOSTNAM.

**HOSTTYP=\*ADD**

Die angegebenen Hosts werden in die bestehende ACCEPT-Liste für den Drucker aufgenommen, von deren Hosts Druckaufträge bearbeitet werden.

**HOSTTYP=\*REMOVE**

Die angegebenen Hosts werden aus der bestehenden ACCEPT-Liste für den Drucker gelöscht, von deren Hosts Druckaufträge bearbeitet werden.

**HOSTTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht veraendert.

**HOSTTYP=\*CLIST**

SPOOLOUT-Aufträge von den in der Liste angegebenen Hosts werden verarbeitet. Alle Elemente der Liste müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**HOSTNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

An dem angegebenen Drucker werden nur Druckaufträge von Hosts dieser ACCEPT-Liste bearbeitet.

**PSAMPLE=\*NO / \*YES / \*UNCHANGED**

Bestimmt, ob für die SPOOLOUT-Aufträge, die unter Berücksichtigung der Zuweisungs-Operanden zur Druckausgabe gelangen, ein Probedruck durchgeführt werden soll. Für Drucker des Typs LP65, für PCL-Drucker und für SPS-Drucker wird PRINT-SAMPLE=\*YES abgewiesen.

Für einen Probedruck auf HP- oder HP90-Drucker darf kein Zeichensatz mit Proportionalschrift verwendet werden. Die Zeichen der auszugebenden Datei werden durch '\*' und '0' ersetzt und sind in Proportionalschrift unterschiedlich breit.

**PSAMPLE=\*NO**

Probedruck wird nicht durchgeführt.

**PSAMPLE=\*YES**

Probedruck wird durchgeführt.

**PSAMPLE=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**TRACE=\*NO / COMPLETE / STATUS / \*BLOCK\_CONTROL / \*UNCHANGED**

Schaltet die Ablaufverfolgung ein oder aus. Name der Trace-Datei:

- für SPOOL:  
\$SYSSPOOL.SYSTRC.SPOOL.<dev-mnemonic>.<yyyy-mm-dd>.<hh-mm>
- für SPS:  
\$SYSSPOOL.SYSTRC.SPS.<dvcname>.<yyyy-mm-dd>.<hhmmss>.<W/S>  
(W für Winter/S für Sommer)
- für RSO:  
\$SYSSPOOL.SYSTRC.RSO.<dvcname>.<yyyy-mm-dd>.<ss.mm>

**TRACE=\*NO**

Der Ablauf soll nicht protokolliert werden.

**TRACE=COMPLETE**

Der Ablauf soll vollständig protokolliert werden.

**TRACE=STATUS**

Nur für SPS-Drucker: Ablauf- und Statusdaten sollen protokolliert werden.

**TRACE=\*BLOCK\_CONTROL**

Nur für SPS-Drucker: Ablauf-, Status- und Block-Kontrolldaten sollen protokolliert werden.

**TRACE=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**USERTYP=\*ALL / \*EXCEPT / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED / \*CLIST**

Legt den Typ der Benutzerkennungen fest, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**USERTYP=\*ALL**

Unabhängig von der Benutzerkennung können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**USERTYP=\*EXCEPT**

Unter den angegebenen Benutzerkennungen können SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden.

**USERTYP=\*LIST**

Liste der Benutzerkennungen, die im Operanden USERNAM angegeben sind, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**USERTYP=\*ADD**

Benutzerkennungen, die zu einer bestehenden Liste von Benutzerkennungen hinzugefügt werden sollen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**USERTYP=\*REMOVE**

Benutzerkennungen, die aus einer bestehenden Liste von Benutzerkennungen gelöscht werden sollen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**USERTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht veraendert.

**USERTYP=\*CLIST**

Liste der Benutzerkennungs-Typen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können. Alle Elemente der Liste müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**USERNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Liste der Benutzerkennungen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*ALL / \*EXCEPT / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED**

SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*ALL**

Unabhängig von der SPOOLOUT-Klasse können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**CLASTYP=\*EXCEPT**

SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden können.

**CLASTYP=\*LIST**

Liste der SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*ADD**

SPOOLOUT-Klassen, die zu einer bestehenden Liste von SPOOLOUT-Klassen hinzugefügt werden sollen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*REMOVE**

SPOOLOUT-Klassen, die aus einer bestehenden Liste von SPOOLOUT-Klassen gelöscht werden sollen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**CLASTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht veraendert.

**CLASNAM=array(16): <var:int:1>/<integer 1..255>**

Liste der SPOOLOUT-Klassen, aus denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*ALL / \*EXCEPT / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED / \*CLIST**

Auftragsnamen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*ALL**

Unabhängig vom Auftragsnamen können SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**SPNATYP=\*EXCEPT**

Unter den angegebenen Auftragsnamen können SPOOLOUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden.

**SPNATYP=\*LIST**

Liste der Auftragsnamen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*ADD**

Auftragsnamen, die zu einer bestehenden Liste von Auftragsnamen hinzugefügt werden sollen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*REMOVE**

Auftragsnamen, die aus einer bestehenden Liste von Auftragsnamen gelöscht werden sollen, unter denen SPOOLOUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**SPNATYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**SPNATYP=\*CLIST**

Liste von Auftragsnamen, unter denen SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden können. Alle Elemente der Liste müssen vom Typ *c-string* sein, mit einer maximalen Länge von 8 Zeichen (Länge ohne Anführungszeichen).

**SPANAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Liste der Auftragsnamen, unter denen SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ACCTYP=\*ALL / \*EXCEPT / \*LIST / \*ADD / \*REMOVE / \*UNCHANGED**

Abrechnungsnummern der SPOOL-OUT-Aufträge, die auf den angegebenen Druckern verarbeitet werden können.

**ACCTYP=\*ALL**

Unabhängig von der Abrechnungsnummer können SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden.

**ACCTYP=\*EXCEPT**

Mit den angegebenen Abrechnungsnummern können SPOOL-OUT-Aufträge auf den angegebenen Druckern nicht verarbeitet werden.

**ACCTYP=\*LIST**

Liste der Abrechnungsnummern, mit denen SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ACCTYP=\*ADD**

Abrechnungsnummern, die zu einer bestehenden Liste von Abrechnungsnummern hinzugefügt werden sollen, mit denen SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ACCTYP=\*REMOVE**

Abrechnungsnummern, die aus einer bestehenden Liste von Abrechnungsnummern gelöscht werden sollen, mit denen SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ACCTYP=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**ACCNAM=array(16): <var:char:8> / <c-string 1..8: c-string 1..8>**

Liste der Abrechnungsnummern, mit denen SPOOL-OUT-Aufträge auf den angegebenen Druckern verarbeitet werden können.

**ROT=\*ANY / \*NO / \*YES / \*MANUAL / \*UNCHANGED**

Bestimmt, ob auf den angegebenen Druckern SPOOL-OUT-Aufträge, die das Seitendrehmodul benötigen, verarbeitet werden können.

**ROT=\*ANY**

Unabhängig davon, ob sie das Seitendrehmodul benötigen, können auf den angegebenen Druckern SPOOL-OUT-Aufträge verarbeitet werden.



**ROT=\*NO**

Auf den angegebenen Druckern können nur SPOOLOUT-Aufträge verarbeitet werden, die das Seitendrehmodul nicht benötigen.

**ROT=\*YES**

Auf den angegebenen Druckern können nur SPOOLOUT-Aufträge verarbeitet werden, die das Seitendrehmodul benötigen.

**ROT=\*MANUAL**

Der Operator kann hier, im Unterschied zum Wert \*NO, manuell über Hardware-Schalter das Seitendrehmodul ansprechen. In diesem Fall werden alle Seiten gedreht ausgedruckt.

**ROT=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht veraendert.

**TWO\_UP=\*ANY / \*YES / \*NO / \*MODE-1 / \*MODE-2 / \*UNCHANGED**

Nur für lokalen SPOOL.

Legt für SPOOLOUT-Aufträge auf HP90-PRINTER die druckerspezifische Eigenschaft TWO-UP-PROCESSING fest:

Es wird festgelegt, ob und in welcher Abfolge zwei nebeneinander liegende Seiten ausgegeben werden sollen. Eine ausführliche Beschreibung der unterschiedlichen 'TWO-UP'-Modi finden Sie im Handbuch „[SPOOL \(BS2000/OSD\)](#)“.

**FOB=(type, low, high)**

Bestimmt, ob auf den angegebenen Druckern Druckaufträge, die ein FOB-Datendia verwenden, verarbeitet werden können.

**type: \*ANY / \*ONLY / \*NO / \*RANGE / \*UNCHANGED**

Art der Druckaufträge, die auf den angegebenen Druckern verarbeitet werden können.

**type: \*ANY**

Druckaufträge mit und ohne FOB-Datendia können auf den angegebenen Druckern verarbeitet werden. Der maximal mögliche FORMS-OVERLAY-BUFFER wird vom korrespondierenden Standard-Geräte-Eintrag entnommen.

**type: \*ONLY**

Nur Druckaufträge, die ein FOB-Datendia verwenden, können auf den angegebenen Druckern verarbeitet werden. Der maximal mögliche FORMS-OVERLAY-BUFFER wird vom korrespondierenden Standard-Geräte-Eintrag entnommen.

**type: \*NO**

Druckaufträge, die ein FOB-Datendia verwenden, können auf den angegebenen Druckern nicht verarbeitet werden.

**type: \*RANGE**

Nur Druckaufträge, die ein FOB-Datendia verwenden, dessen Größe innerhalb des angegebenen Intervalls liegt, können auf den angegebenen Druckern verarbeitet werden.

**type: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**low: \*STD / <var:int:2> / <integer 0..32767> / \*UNCHANGED**

Untergrenze des Intervalls (Angabe in Unterzeilen). \*STD entspricht dabei dem Wert 0.

**high: \*STD / <var:int:2> / <integer 0..32767> / \*UNCHANGED**

Obergrenze des Intervalls für HP90-Drucker: 4032 Unterzeilen. Obergrenze des Intervalls für HP-PRINTER: 672 Unterzeilen. \*STD entspricht dabei dem Wert 4032.

**PRIO=(type, from, to)**

Prioritäten der Druckaufträge, die auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL / \*RANGE / \*UNCHANGED**

Art der Druckaufträge, die auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL**

Druckaufträge können unabhängig von ihrer Priorität auf den angegebenen Druckern verarbeitet werden.

**type: \*RANGE**

Druckaufträge, deren Priorität innerhalb des angegebenen Intervalls liegt, können auf den angegebenen Druckern verarbeitet werden.

**type: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**from: \*MIN / <var: int: 1> / <integer 30..255> / \*UNCHANGED**

Untergrenze des Intervalls. \*MIN entspricht dabei dem Wert 30.

**to: \*MAX / <var: int: 1> / <integer 30..255> / \*UNCHANGED**

Obergrenze des Intervalls.\*MAX entspricht dem Wert 255.

**CHARSN=(type, low, high)**

Anzahl der benötigten Zeichensätze, mit denen Druckaufträge auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL /\*ONE / \*RANGE / \*UNCHANGED**

Zeichensätze, mit denen Druckaufträge auf den angegebenen Druckern verarbeitet werden können.

**type: \*ALL**

Unabhängig von der Anzahl der Zeichensätze können Druckaufträge auf den angegebenen Druckern verarbeitet werden. Die maximal mögliche Anzahl der benötigten Zeichensätze wird vom korrespondierenden Standard-Geräte-Eintrag entnommen.

**type: \*ONE**

Nur Druckaufträge, die nicht mehr als einen Zeichensatz benötigen, können auf den angegebenen Druckern verarbeitet werden.

**type: \*RANGE**

Druckaufträge, deren Anzahl benötigter Zeichensätze innerhalb des angegebenen Intervalls liegt, können auf den angegebenen Druckern verarbeitet werden. Während der Initialisierung eines Geräts wird geprüft, wieviele Zeichensätze auf dem Gerät geladen werden können.

Ist dieser Wert kleiner als einer der beiden angegebenen Werte, wird entweder eine Meldung an die Bedienstation geschickt und der Wert für *high* angepasst (wenn  $low \leq$  Anzahl der Zeichensätze, die auf dem Gerät geladen werden können), oder das Kommando wird abgewiesen (wenn  $low >$  Anzahl der Zeichensätze, die auf dem Gerät geladen werden können).

**type: \*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**low: \*MIN / <var:int:2> / <integer 1..32767> / \*UNCHANGED**

Untergrenze des Intervalls. \*MIN entspricht dabei dem Wert 1.

**high: \*MAX / <var:int:2> / <integer 1..32767> / \*UNCHANGED**

Obergrenze des Intervalls. \*MAX entspricht dabei dem Wert 64.

**EXITR=\*ACTIVE / \*NOT-ACTIVE / \*UNCHANGED**

Gibt an, ob EXIT-Routinen beim SPOOLOUT aufgerufen werden sollen.

**EXITR=\*ACTIVE**

EXIT-Routinen sollen beim SPOOLOUT aufgerufen werden.

**EXITR=\*NOT-ACTIVE**

EXIT-Routinen sollen beim SPOOLOUT nicht aufgerufen werden.

**EXITR=\*UNCHANGED**

Vorgabewert dieses Operanden. Die aktuelle Einstellung wird nicht verändert.

**RVSION=<var: int:1> / <integer 1..255>**

Nummer des Bearbeitungsstandes. Die angegebene Nummer muss mit der des gerade aktuellen Bearbeitungsstandes übereinstimmen

## Returncodes

(SC2)	SC1	Maincode	Bedeutung
00	00	0000	Kein Fehler
02	00	xxxx	Warnung - Drucker existiert nicht oder Ausführung ist nicht erlaubt
02	01	FFFF	Ungültige Parameterlisteara
01	01	xxxx	Ungültiger DEVICE-Operand
02	01	xxxx	Ungültiger DESTINATION-Operand
03	01	xxxx	Ungültiger FORM-NAME-Operand
04	01	xxxx	Ungültiger OVERLAY-Operand
05	01	xxxx	Ungültiger HOST-NAME-Operand
06	01	xxxx	Ungültiger PAGE-PRINTER-OUTPUT-Operand
07	01	xxxx	Ungültiger PRINT-SAMPLE-Operand
08	01	xxxx	Ungültiger TRACE-Operand
09	01	xxxx	Ungültiger ACCESS-Operand
0A	01	xxxx	Ungültiger USERID-Operand
0B	01	xxxx	Ungültiger SPOOLOUT-CLASS-Operand
0C	01	xxxx	Ungültiger JOB-NAME-Operand
0D	01	xxxx	Ungültiger ACCOUNT-Operand
0E	01	xxxx	Ungültiger ROTATION-Operand
0F	01	xxxx	Ungültiger TWO-UP -Operand
10	01	xxxx	Ungültiger FORM-OVERLAY-BUFFER-Operand
11	01	xxxx	Ungültiger PRIORITY-Operand
12	01	xxxx	Ungültiger CHARACTER-SET-NUMBER-Operand
13	01	xxxx	Ungültige Adresse der RSO-Parameterlistenerweiterung
14	01	xxxx	Ungültiger REVISION-Operand
15	01	xxxx	Ungültiger EXIT-Operand
00	20	xxxx	Systemfehler
02	40	xxxx	Keine Berechtigung
04	40	xxxx	Operandenwertfehler
05	40	xxxx	Parameterlistenversion wird nicht unterstützt
02	80	xxxx	RSO-Subsystem ist nicht geladen
05	80	xxxx	SPS-Subsystem ist nicht geladen
06	80	xxxx	Dprintcl-Subsystem ist nicht geladen
07	80	xxxx	Dprintcm-Subsystem ist nicht geladen
08	80	xxxx	Dprintsv-Subsystem ist nicht geladen
FF	FF	FFFF	Spool-Subsystem ist nicht geladen

*Hinweis*

Die vier Zeichen xxxx in der Maincode-Spalte bezeichnen eine Meldungskennung. Um eine ausführlichere Information über einen Fehler zu erhalten, geben Sie das Kommando HELP-MSG SCPxxxx oder SPSxxxx im SYSTEM MODUS ein.

**Beschreibung der Parameterliste**

```

*   _dev_type_set
      AIF ('&EQUATES' NE 'YES').Q01001
SUPANYL EQU 1 local device
SUPRSO EQU 2 RSO device
.Q01001 ANOP
*
*   parameter list description
SUP_MDL DS 0F
SUPHDR FHDR MF=(C,SUP),EQUATES=NO Standard header
*   main return codes
      AIF ('&EQUATES' NE 'YES').Q01002
SUPRCNO EQU 0 subcode1 = 00 : no error
SUPWARN EQU 2 printer not found or
* processing not permitted
SUPVALM EQU 2 subcode1 = 01 : pl area
* invalid - maincode = FFFF
SUPPARM EQU 1 subcode1 = 01 : wrong
* parameter
SUPDEVI EQU 1 subcode1 = 01 : device
SUPDEST EQU 2 subcode1 = 01 : destination
SUPFONA EQU 3 subcode1 = 01 : form name
SUPOVRL EQU 4 subcode1 = 01 : overlay
SUPHOST EQU 5 subcode1 = 01 : host
SUPPPOU EQU 6 subcode1 = 01 : page printer
* output
SUPPRSA EQU 7 subcode1 = 01 : print sample
SUPTRAE EQU 8 subcode1 = 01 : trace
SUPACCE EQU 9 subcode1 = 01 : access
SUPUSER EQU 10 subcode1 = 01 : userid
SUPSPCL EQU 11 subcode1 = 01 : spoolout
* class
SUPJNAM EQU 12 subcode1 = 01 : job name
SUPACCN EQU 13 subcode1 = 01 : account
SUPROTA EQU 14 subcode1 = 01 : rotation
SUPTWOU EQU 15 subcode1 = 01 : two-up
SUPFOB EQU 16 subcode1 = 01 : fob
SUPPRIO EQU 17 subcode1 = 01 : priority
SUPCSNU EQU 18 subcode1 = 01 : char_set_num
SUPRSOL EQU 19 subcode1 = 01 : addr of rso
* pl
SUPREVI EQU 20 subcode1 = 01 : revision
SUPEXTT EQU 21 subcode1 = 01 : exit
SUPSYSE EQU 0 subcode1 = 20 : system error
SUPPRIV EQU 2 subcode1 = 40 : no
* authorization
SUPVERR EQU 4 operand value error

```

```

SUPVERE    EQU    5                p/l version not supported
SUPNOSS    EQU    0                subcode1 = 80 : subsystem not
*                                                loaded
SUPNRSO    EQU    2                rso not loaded
SUPNDCL    EQU    6                dprintcl not loaded
SUPNDCM    EQU    7                dprintcm not loaded
SUPNDSV    EQU    8                dprintsv not loaded
SUPNSPO    EQU    255             subcode1 = 255: spool not
*                                                loaded
.Q01002    ANOP
*
*
SUPADVNM    DS      0XL132         device name
SUPAFC8B    DS      FL1
*   _flag_set
      AIF ('&EQUATES' NE 'YES').Q01003
SUPFALL     EQU    0                _type = *all
SUPFNON     EQU    1                _type = *none
SUPFLOC     EQU    2                _type = *local
SUPFREM     EQU    3                _type = *remote
SUPFCEN     EQU    4                _type = *central
SUPFDEV     EQU    64              _type = *device
SUPFMDV     EQU    128             _type = *manag_dev
SUPFWLD     EQU    5                _type = *wildcard
SUPFLST     EQU    6                _type = *list
SUPFSTD     EQU    7                _type = *std
SUPFOWN     EQU    8                _type = *own
SUPFHOM     EQU    9                _type = *home
SUPFIDE     EQU    10              _type = *ident
SUPFANY     EQU    11              _type = *any
SUPFONL     EQU    12              _type = *only
SUPFRNG     EQU    13              _type = *range
SUPFONE     EQU    14              _type = *one
SUPFNO      EQU    15              _type = *no
SUPEQU     EQU    16              _type=*equivalent
SUPEQUX    EQU    17              _type=*equivalent_ex
SUPEXCE    EQU    18              _type=*except
SUPALLO     EQU    19              _type=*allowed
SUPFYES    EQU    20              _type = *yes
SUPFCOM     EQU    21              _type = *complete
SUPFSTA     EQU    22              _type = *status
SUPFBLK     EQU    23              _type = *block_control
SUPFMAN     EQU    24              _type = *manual
SUPFM01     EQU    25              _type = *mode-1
SUPFM02     EQU    26              _type = *mode2
SUPFACT     EQU    27              _type = *active
SUPFNOA     EQU    28              _type = *no-active
SUPFUNC     EQU    29              _type = *unchanged

```

```

SUPFADD    EQU    30                _type = *add
SUPFREV    EQU    31                _type = *remove
SUPFUPD    EQU    32                _type = *update
SUPFMPR    EQU    30                _type = *minprio
SUPFMAP    EQU    255               _type = *maxprio
SUPFMCH    EQU    1                 _type = *minchars
SUPFMXC    EQU    64                _type = *maxchars
.Q01003    ANOP
*
SUPAUC8B   DS     XL3
SUPAEC8B   DS     8CL8
            AIF   ('&EQUATES' NE 'YES').Q01004
SUPAEC8B#  EQU    8
.Q01004    ANOP
SUPAEC82   DS     8CL8
            AIF   ('&EQUATES' NE 'YES').Q01005
SUPAEC82#  EQU    8
.Q01005    ANOP
*
*
SUPDDEST   DS     0XL132            destination
SUPDFC8    DS     FL1
SUPDUC8    DS     XL3
SUPDEC8    DS     16CL8
            AIF   ('&EQUATES' NE 'YES').Q01006
SUPDEC8#   EQU    16
.Q01006    ANOP
*
*
SUPAFRMN   DS     0XL100            form name
SUPAFC6    DS     FL1
SUPAUC6    DS     XL3
SUPAEC6    DS     16CL6
            AIF   ('&EQUATES' NE 'YES').Q01007
SUPAEC6#   EQU    16
.Q01007    ANOP
*
*
SUPAFOVL   DS     0XL36            form overlay
SUPAFC2    DS     FL1
SUPAUC2    DS     XL3
SUPAEC2    DS     16CL2
            AIF   ('&EQUATES' NE 'YES').Q01008
SUPAEC2#   EQU    16
.Q01008    ANOP
*
*
SUPBHOST   DS     0XL132            host name

```

```

SUPBFC8    DS    FL1
SUPBUC8    DS    XL3
SUPBEC8    DS    16CL8
            AIF  ('&EQUATES' NE 'YES').Q01009
SUPBEC8#   EQU   16
.Q01009    ANOP
*
SUPPPO     DS    FL1                page printer output
SUPPSAM    DS    FL1                print sample
SUPTRAC    DS    FL1                trace
SUPUN1     DS    XL1
*
SUPEACCS   DS    0XL20                allowed accesses
SUPEFC4    DS    FL1
SUPEUC4    DS    XL3
SUPEEC4    DS    4CL4
            AIF  ('&EQUATES' NE 'YES').Q01010
SUPEEC4#   EQU   4
.Q01010    ANOP
*
*
SUPCUSRD   DS    0XL132                userid
SUPCFC8    DS    FL1
SUPCUC8    DS    XL3
SUPCEC8    DS    16CL8
            AIF  ('&EQUATES' NE 'YES').Q01011
SUPCEC8#   EQU   16
.Q01011    ANOP
*
*
SUPASPCL   DS    0XL20                spoolout class
SUPAFI1    DS    FL1
SUPAUC1    DS    XL3
SUPAEI1    DS    16X
*
*
SUPESPNA   DS    0XL132                spoolout name
SUPEFC8    DS    FL1
SUPEUC8    DS    XL3
SUPEEC8    DS    16CL8
            AIF  ('&EQUATES' NE 'YES').Q01012
SUPEEC8#   EQU   16
.Q01012    ANOP
*
*
SUPFACNT   DS    0XL132                account
SUPFFC8    DS    FL1
SUPFUC8    DS    XL3

```



```

SUPFEC8    DS    16CL8
           AIF    ('&EQUATES' NE 'YES').Q01013
SUPFEC8#   EQU    16
.Q01013    ANOP
*
SUPROT     DS    FL1                rotation
SUTUP      DS    FL1                two-up processing
SUPUN2     DS    XL2
*
SUPMFOB    DS    0XL8                form-overlay-buffer
SUPMFR2    DS    FL1
SUPMUN1    DS    XL1
SUPMLR2    DS    H
SUPMHR2    DS    H
SUPMUR2    DS    XL2
*
*
SUPOPRI0   DS    0XL4                priority
SUPOFR1    DS    FL1
SUPOLR1    DS    X
SUPOHR1    DS    X
SUPOUR1    DS    XL1
*
*
SUPACHAR   DS    0XL8                character-set-number
SUPAFR2    DS    FL1
SUPAUN1    DS    XL1
SUPALR2    DS    H
SUPAHR2    DS    H
SUPAUR2    DS    XL2
*
*
SUPRSOP    DS    A                address of rso_p1
SUPPTRRES1 DS    A                address of reserved1
SUPPTRRES2 DS    A                address of reserved2
SUPPTRRES3 DS    A                address of reserved3
SUPREVIS   DS    X                revision number
SUPUN4     DS    XL3
SUPEXIT    DS    FL1                exit-processing
SUPSP1     DS    AL1                specified1
           AIF    ('&EQUATES' NE 'YES').Q01014
SUPSP11    EQU    X'80'            device type
SUPSP12    EQU    X'40'            rso param
SUPSP13    EQU    X'20'            destination type
SUPSP14    EQU    X'10'            form type
SUPSP15    EQU    X'08'            overlay type
SUPSP16    EQU    X'07'            unused
.Q01014    ANOP
SUPSP2     DS    AL1                specified2
    
```

```

        AIF ('&EQUATES' NE 'YES').Q01015
SUPSP21 EQU X'80'          host type
SUPSP22 EQU X'40'          page pr. output
SUPSP23 EQU X'20'          print sample
SUPSP24 EQU X'10'          trace
SUPSP25 EQU X'08'          userid type
SUPSP26 EQU X'07'          unused
.Q01015 ANOP
SUPSP3  DS AL1             specified3
        AIF ('&EQUATES' NE 'YES').Q01016
SUPSP31 EQU X'80'          spoolout class type
SUPSP32 EQU X'40'          spoolout name type
SUPSP33 EQU X'20'          spoolout name list
SUPSP34 EQU X'10'          account type
SUPSP35 EQU X'08'          rotation
SUPSP36 EQU X'04'          two-up processing
SUPSP37 EQU X'02'          charsetnum low
SUPSP38 EQU X'01'          charsetnum high
.Q01016 ANOP
SUPSP4  DS AL1             specified4
        AIF ('&EQUATES' NE 'YES').Q01017
SUPSP41 EQU X'80'          fob type
SUPSP42 EQU X'40'          fob range low
SUPSP43 EQU X'20'          fob range high
SUPSP44 EQU X'10'          priority
SUPSP45 EQU X'08'          priority from
SUPSP46 EQU X'04'          priority to
SUPSP47 EQU X'02'          charsetnum
SUPSP48 EQU X'01'          exit routines
.Q01017 ANOP
SUPUN3  DS XL43
        AIF ('&EQUATES' NE 'YES').Q01018
SUP#    EQU *-SUPHDR
.Q01018 ANOP

```

---

## 6 Makros für virtuelle Drucker

Virtuelle Drucker sind in den Spool & Print Services Objekte, die wie reale Geräte behandelt werden, aber dennoch nur als Anwendung existieren, die über einen Gerätenamen angesprochen wird. Wie ein reales Gerät kann ein virtueller Drucker Aufträge vom SPOOL entgegennehmen. Ein virtueller Drucker gibt diese Aufträge entweder sofort an eine synchron arbeitende Anwendung weiter, oder macht die Daten des Auftrags für eine asynchron arbeitende Anwendung zugänglich.

Zur Kommunikation der Anwendung mit dem virtuellen Drucker im synchronen Betrieb steht eine Programmschnittstelle zur Verfügung. Diese Programmierschnittstelle besteht aus den folgenden Makros:

### „SPVDINI - Dialog mit virtuellem Drucker initialisieren“ auf Seite 503

SPVDINI initialisiert den Dialogmechanismus zwischen dem virtuellen Drucker und der Benutzeranwendung. Der Dialogmechanismus muss einmal während des Starts der Anwendung mit SPVDINI initialisiert werden.

### „SPVDRCV - Anwendungstask empfangsbereit setzen“ auf Seite 505

SPVDRCV empfängt Daten von dem zur Anwendung gehörenden virtuellen Drucker. SPVDRCV ist eine synchrone Schnittstelle, was bedeutet, dass SPVDRCV auf Daten vom angeschlossenen virtuellen Drucker wartet, sobald die Anwendung diese Funktion aufgerufen hat. Der Dialog mit dem virtuellen Drucker muss mit SPVDINI initialisiert worden sein, bevor SPVDRCV aufgerufen werden kann.

### „SPVDRET - Informationen über den aktuellen Auftrag zum virtuellen Drucker senden“ auf Seite 507

SPVDRET gibt das Bearbeitungsergebnis der letzten Anfrage an den angeschlossenen virtuellen Drucker zurück. Dazu muss der vorangegangene Aufruf von SPVDRCV erfolgreich beendet worden sein. SPVDRET muss zudem aufgerufen werden, sobald die Anwendung einen Auftrag komplett bearbeitet hat.

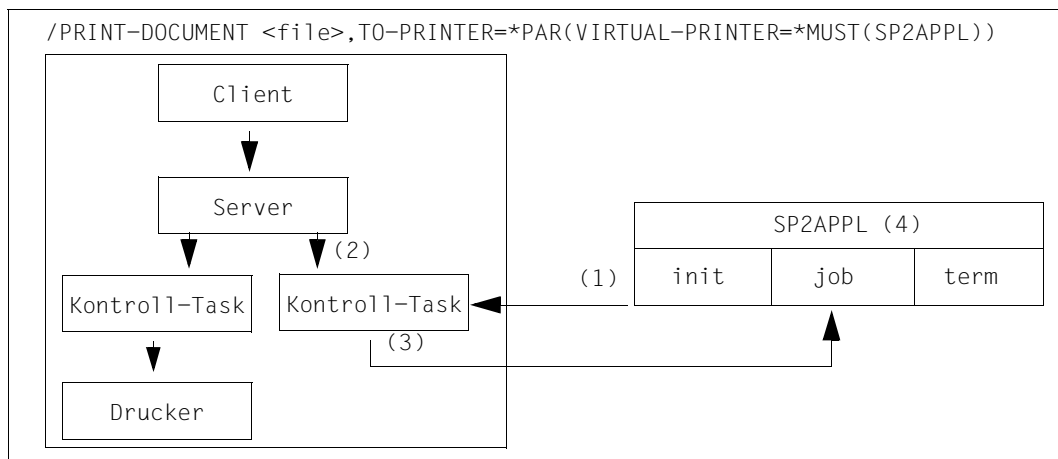
### „SPVDEND - Dialogverbindung beenden“ auf Seite 509

SPVDEND beendet die Verbindung zu dem angeschlossenen virtuellen Drucker.

## 6.1 Einsatz virtueller Drucker im synchronen Betrieb

Folgendes Beispiel zeigt den Einsatz virtueller Drucker im synchronen Betrieb.

Ein Drucker SP2APPL ist in der SPOOL-Parameterdatei mit dem Gerätetyp \*VIRTUAL in Verbindung mit einem Programm eingetragen worden. Dieses Programm realisiert eine besondere Behandlung der Aufträge und die Schnittstelle für den Dialog mit der Kontrolltask des virtuellen Druckers.



- (1) Mit dem Start des virtuellen Druckers wird die Kontrolltask erzeugt. Während der Initialisierung prüft die Kontrolltask die Definition von SP2APPL und liest die Beschreibung dieses Programms. Ein ENTER-JOB dieser Datei wird ausgeführt. Diese Task läuft unter \$SYSSPOOL mit der Auftragsklasse \$SYSJC (wenn im Prozedurnamen keine Benutzerkennung angegeben wurde), der vorgebenen Priorität und dem Auftragsnamen SP2APPL, dem einmaligen Kennzeichen für die Logon-Prozedur.

Der Benutzer kann die Logon-Prozedur nach seinen Wünschen programmieren und gestalten. Davon ist ausgenommen, dass während der Stapelverarbeitung ein TU-Programm geladen und gestartet werden muss, um den Dialog mit der Kontrolltask über die Programmschnittstelle zu etablieren.

- (2) Jedes Kommando PRINT-DOCUMENT VIRTUAL-PRINTER=\*MUST(SP2APPL) oder jeder Druckauftrag, der den Auswahlkriterien entspricht, die beim Start des virtuellen Geräts angegeben wurden, wird zu einem Auftrag für die Kontrolltask des Geräts SP2APPL.
- (3) Die Kontrolltask übergibt den Auftrag an die Anwendung mit einem Aktionscode "job", den Auftragsattributen und wahlweise einer Zeichenkette, die bei der Auftragsvergabe erzeugt wurde.

- (4) Mit diesen Informationen kann die Anwendung den Auftrag bearbeiten. Danach informiert die Anwendung die Kontrolltask über das Ende der Auftragsbearbeitung und wie mit dem Auftrag weiter zu verfahren ist.

Wenn der virtuelle Drucker beendet wird, erhält die Anwendung den Aktionscode "term". Für den Dialog zwischen Anwendung und Kontrolltask steht eine spezielle Programmschnittstelle bereit, die für den Nachrichtenaustausch zwischen den Partnern einen FITC-Mechanismus nutzt.

## 6.2 Programmschnittstelle für den synchronen Betrieb

Im synchronen Betrieb kommuniziert ein virtueller Drucker mit einer Benutzeranwendung. Um diese Kommunikation zu erleichtern, steht eine Programmschnittstelle bereit, die für den Nachrichtenaustausch einen FITC (Fast Inter Task Communication) - Mechanismus benutzt. Diese Schnittstelle umfasst vier Funktionen:

- SPVDINI** initialisiert den Dialogmechanismus zwischen dem virtuellen Drucker und der Benutzeranwendung. Der Dialogmechanismus muss einmal während des Starts der Anwendung mit SPVDINI initialisiert werden.
- SPVDRCV** empfängt Daten von dem zur Anwendung gehörenden virtuellen Drucker. SPVDRCV ist eine synchrone Schnittstelle, was bedeutet, dass SPVDRCV auf Daten vom angeschlossenen virtuellen Drucker wartet, sobald die Anwendung diese Funktion aufgerufen hat. Der Dialog mit dem virtuellen Drucker muss mit SPVDINI initialisiert worden sein, bevor SPVDRCV aufgerufen werden kann.
- SPVDRET** gibt das Bearbeitungsergebnis der letzten Anfrage an den angeschlossenen virtuellen Drucker zurück. Dazu muss der vorangegangene Aufruf von SPVDRCV erfolgreich beendet worden sein. SPVDRET muss zudem aufgerufen werden, sobald die Anwendung einen Auftrag komplett bearbeitet hat.
- SPVDEND** beendet die Verbindung zu dem angeschlossenen virtuellen Drucker.

Ein Beispiel für eine Benutzeranwendung, die die beschriebenen Programmschnittstellen nutzt, um mit einem virtuellen Drucker im synchronen Betrieb zu kommunizieren, finden Sie ab [Seite 516](#).

## SPVDINI - Dialog mit virtuellem Drucker initialisieren

Initialisiert den Dialogmechanismus zwischen dem virtuellen Drucker und der Benutzeranwendung. Der Dialogmechanismus muss einmal während des Starts der Anwendung mit SPVDINI initialisiert werden.

### Format

Operation	Operanden
SPVDINI	MF=C / D / E / L ,VARIANT=001 / <c-string 3..3> [,MACID=PDI / macid] [,PREFIX=S / p]

### Operandenbeschreibung

#### MF=C / D / E / L

Die Formen des MF-Operanden sind detailliert im [Abschnitt „Typen von Makroaufrufen“ auf Seite 602](#) beschrieben. In allen Makroaufrufen, die sich durch den MF-Operanden unterscheiden (MF=L/E/D/C/M), muss der Versionsoperand den gleichen Wert haben.

#### VARIANT=001 / <c-string 3..3>

Bezeichnet die Variante der generierten Parameterliste.

#### MACID=PDI / macid

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt jeweils das zweite bis einschließlich dritte Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

#### MACID=PDI

Voreinstellung für diesen Operanden.

#### MACID=macid

Ein oder zwei Zeichen lange Zeichenfolge, die jeweils das zweite bis dritte Zeichen der generierten Feldnamen und Equates festlegt.

#### PREFIX=S / p

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt das jeweils erste Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

#### PREFIX=S

Ist das voreingestellte Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen.

**PREFIX=p**

"p" ist ein ein Zeichen langes Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen sollen.

**PREFIX=\***

Es wird kein Präfix generiert.

**Returncodes**

SRC2	SRC1	MRC	Bedeutung
00	00	0000	erfolgreich ausgeführt
00	00	0001	Dialog nicht möglich
00	00	0002	nur ein Aufruf durch die Task



## SPVDRCV - Anwendungstask empfangsbereit setzen

Empfängt Daten von dem zur Anwendung gehörenden virtuellen Drucker. SPVDRCV ist eine synchrone Schnittstelle, was bedeutet, dass SPVDRCV auf Daten vom angeschlossenen virtuellen Drucker wartet, sobald die Anwendung diese Funktion aufgerufen hat. Der Dialog mit dem virtuellen Drucker muss mit SPVDINI initialisiert worden sein, bevor SPVDRCV aufgerufen werden kann.

### Format

Operation	Operanden
SPVDRCV	MF = C / D / E / L / M ,VARIANT = <u>001</u> / <c-string 3..3> ,OUTADDR = <var:pointer> ,OUTLGT = <var:int:2> / <integer 1..32000> ,MACID = <u>PDR</u> / <name 1..3> ,PREFIX= <u>S</u> / <name 1..1> ,PARAM = <var:pointer>

### Operandenbeschreibung

#### MF=C / D / E / L / M

Die Formen des MF-Operanden sind detailliert im [Abschnitt „Typen von Makroaufrufen“ auf Seite 602](#) beschrieben. In allen Makroaufrufen, die sich durch den MF-Operanden unterscheiden (MF=L/E/D/C/M), muss der Versionsoperand den gleichen Wert haben.

#### VARIANT=001 / <c-string 3..3>

Bezeichnet die Variante der generierten Parameterliste.

#### OUTADDR=<var: pointer>

Adresse des Ausgabepuffers für die Daten von der Kontrolltask.

#### OUTLGT=<var: int:2> / <integer 1..32000>

Länge des Ausgabepuffers für die Daten von der Kontrolltask.

#### MACID=PDR / <name 1..3>

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt jeweils das zweite bis einschließlich dritte Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

#### MACID=PDR

Voreinstellung für diesen Operanden.

**MACID=<name 1..3>**

Ein bis drei Zeichen lange Zeichenfolge, die das zweite, dritte und vierte Zeichen der generierten Feldnamen und Equates festlegt.

**PREFIX=S / <name 1..1>**

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt das jeweils erste Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

**PREFIX=S**

Ist das voreingestellte Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen.

**PREFIX=<name 1..1>**

<name 1..1> ist ein ein Zeichen langes Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen sollen.

**PARAM=<var:pointer>**

Adresse der Parameterliste.

**Verwendung der Register**

Register 1 wird zur Adressierung der Parameterliste verwendet.

R15 kann nur geändert werden, wenn beim Aufruf des Makros das SPOOL-Subsystem nicht vorhanden ist.

**Returncodes**

SRC2	SRC1	MRC	Bedeutung
00	00	0000	erfolgreich ausgeführt
00	00	0001	Dialog nicht möglich
00	00	0002	SPVDINI-Aufruf erwartet - keine Daten empfangen
00	00	0004	SPVDRET-Aufruf erwartet - keine Daten empfangen
00	00	0008	Arbeit bereits beendet - keine Daten empfangen
00	00	000a	Fehler in der Parameterliste

## SPVDRET - Informationen über den aktuellen Auftrag zum virtuellen Drucker senden

Gibt das Bearbeitungsergebnis der letzten Anfrage an den angeschlossenen virtuellen Drucker zurück. Dazu muss der vorangegangene Aufruf von SPVDRCV erfolgreich beendet worden sein. SPVDRET muss zudem aufgerufen werden, sobald die Anwendung einen Auftrag komplett bearbeitet hat.

### Format

Operation	Operanden
SPVDRET	MF=C / D / E / L / M ,VARIANT= <u>001</u> / <c-string 3..3> ,PACTION=<var: enum-of _post_action_set> ,APPLERR=<var: char:8> / <c-string 8..8> ,OUTADD=<var: pointer> ,OUTLGT=<var: int:2> / <integer 1..32000> ,MACID= <u>PDT</u> / <name 1..3> ,PREFIX= <u>S</u> / <name 1..1> ,PARAM = <var:pointer>

### Operandenbeschreibung

#### MF=C / D / E / L / M

Die Formen des MF-Operanden sind detailliert im [Abschnitt „Typen von Makroaufrufen“ auf Seite 602](#) beschrieben. In allen Makroaufrufen, die sich durch den MF-Operanden unterscheiden (MF=L/E/D/C/M), muss der Versionsoperand den gleichen Wert haben.

#### VARIANT=001 / <c-string 3..3>

Bezeichnet die Variante der generierten Parameterliste.

#### PACTION=<var: enum-of \_post\_action\_set>

Aktion, die vom virtuellen Drucker auf den aktuellen Auftrag angewendet werden soll.

#### APPLERR=<var: char:8> / <c-string 8..8>

Fehler-Code in druckbarer Form, der einen Fehler beschreibt, auf den die Anwendung während der Bearbeitung des aktuellen Auftrags gelaufen ist.

#### OUTADD=<var: pointer>

Adresse des Ausgabepuffers, der eine Beschreibung des Druckauftrags enthält. Diese Beschreibung ist eine SSVSCBE-Parameterliste, der SPOOL Control Block.

**OUTLGT=<var: int:2> / <integer 1..32000>**

Länge des Ausgabepuffers.

**MACID=PDT / <name 1..3>**

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt jeweils das zweite bis einschließlich dritte Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

**MACID=PDT**

Voreinstellung für diesen Operanden.

**MACID=<name 1..3>**

Ein bis drei Zeichen lange Zeichenfolge, die das zweite, dritte und vierte Zeichen der generierten Feldnamen und Equates festlegt.

**PREFIX=S / <name 1..1>**

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt das jeweils erste Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

**PREFIX=S**

Ist das voreingestellte Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen.

**PREFIX=\***

Es wird kein Präfix erzeugt.

**PREFIX=<name 1..1>**

<name 1..1> ist ein ein Zeichen langes Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen sollen.

**PARAM=<var:pointer>**

Adresse der Parameterliste.

**Verwendung der Register**

Register 1 wird zur Adressierung der Parameterliste verwendet.

R15 kann nur geändert werden, wenn beim Aufruf des Makros das SPOOL-Subsystem nicht vorhanden ist.

**Returncodes**

SRC2	SRC1	MRC	Bedeutung
00	00	0000	erfolgreich ausgeführt
00	00	0001	Fehler bei Übergabe von Daten an den virtuellen Drucker

## SPVDEND - Dialogverbindung beenden

Beendet die Verbindung zu dem angeschlossenen virtuellen Drucker.

### Format

Operation	Operanden
SPVDEND	MF=C / D / E / L ,VARIANT=001 / <c-string 3..3> [,MACID=PDE / macid] [,PREFIX=S / p]

### Operandenbeschreibung

#### MF=C / D / E / L

Die Formen des MF-Operanden sind detailliert im [Abschnitt „Typen von Makroaufrufen“ auf Seite 602](#) beschrieben. In allen Makroaufrufen, die sich durch den MF-Operanden unterscheiden (MF=L/E/D/C/M), muss der Versionsoperand den gleichen Wert haben.

#### VARIANT=001 / <c-string 3..3>

Bezeichnet die Variante der generierten Parameterliste.

#### MACID=PDE / macid

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt jeweils das zweite bis einschließlich dritte Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

#### MACID=PDE

Voreinstellung für diesen Operanden.

#### MACID=macid

Ein oder zwei Zeichen lange Zeichenfolge, die jeweils das zweite bis dritte Zeichen der generierten Feldnamen und Equates festlegt.

#### PREFIX=S / p

Wird nur in Verbindung mit MF=C/D/M ausgewertet und legt das jeweils erste Zeichen der Feldnamen und Equates fest, die bei der Makroauflösung im Datenbereich generiert werden.

#### PREFIX=S

Ist das voreingestellte Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen.

#### PREFIX=p

"p" ist ein ein Zeichen langes Präfix, mit dem die vom Assembler generierten Feldnamen und Equates beginnen sollen.

**PREFIX=\***

Es wird kein Präfix generiert.

**Returncodes**

<b>SRC2</b>	<b>SRC1</b>	<b>MRC</b>	<b>Bedeutung</b>
00	00	0000	erfolgreich ausgeführt
00	00	0001	interner Fehler

## 6.3 Kommunikation zwischen virtuellem Drucker und Anwendung

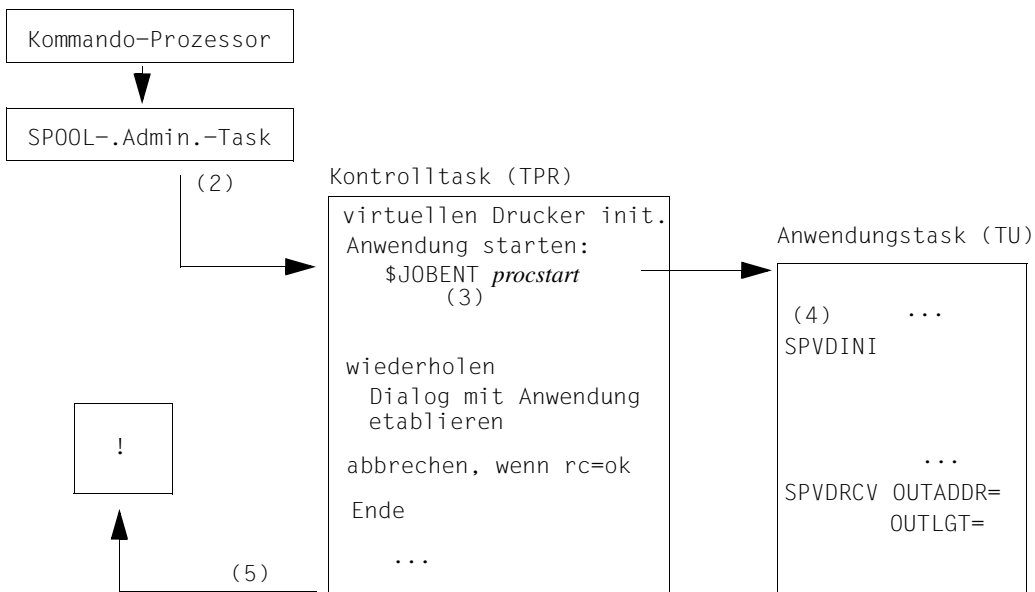
Der virtuelle Drucker läuft unter der Kontrolle einer TPR-Task (Kontrolltask), während die Anwendung unter der Kontrolle einer TU-Task (Anwendungstask) läuft. Die beiden Tasks tauschen Daten aus.

### 6.3.1 Initialisieren des Dialogs

Ein virtueller Drucker *virdev* wurde mit folgender SPSERVE-Anweisung definiert:  
`/ADD-SPOOL-DEVICE virdev, DEVICE-TYPE=*VIRTUAL, DEVICE-ACCESS=*LOCAL-ACCESS(PROGRAM-NAME=procstart)`

Dabei ist *procstart* der Name der Prozedurdatei, die zumindest das Kommando START-PROGRAM für die Anwendung enthält, die mit dem virtuellen Drucker verbunden ist.

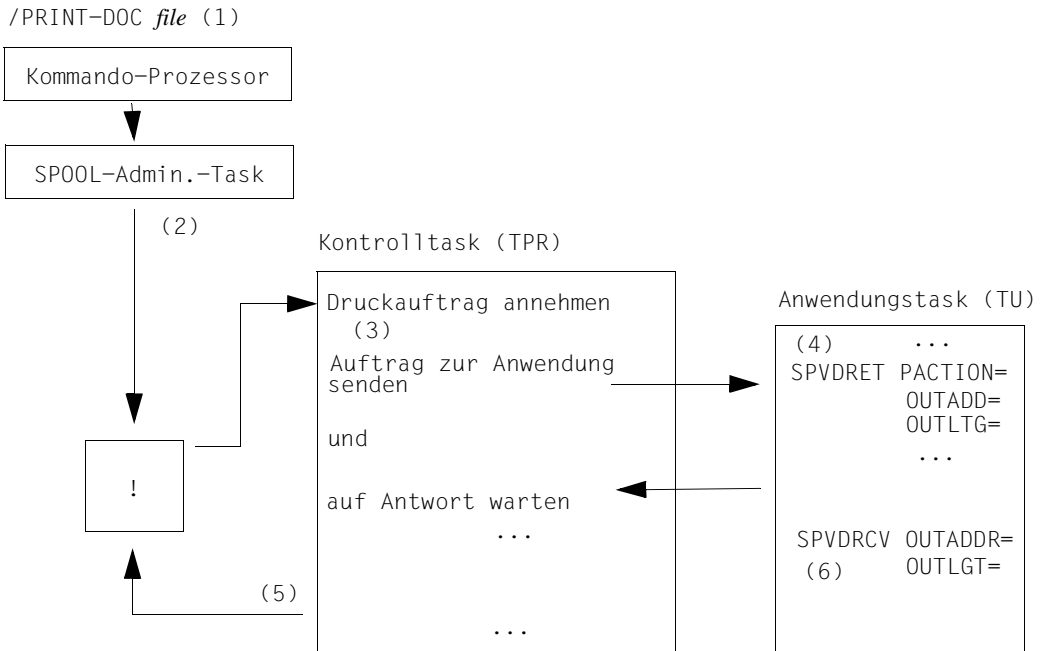
`/START-PRINTER-OUTPUT virdev (1)`



- (1) Mit START-PRINTER-OUTPUT startet der SPOOL-Verwalter den virtuellen Drucker.
- (2) Die SPOOL-Administrator-Task erzeugt die zum virtuellen Drucker gehörende Kontrolltask (TPR-Task).
- (3) Die Kontrolltask erzeugt einen ENTER-Job mit dem Prozedurnamen, den sie in der Gerätedefinition findet und versucht einen Dialog mit der entsprechenden Anwendungstask zu etablieren.
- (4) Die Anwendung muss den Dialog mit SPVDINI initialisieren und sich selbst mit SPVDRCV empfangsbereit setzen. Die Anwendung definiert in diesem Aufruf die Adresse und die Größe eines Speicherbereichs für den Empfang von Daten vom virtuellen Drucker.
- (5) Nachdem der Dialog erfolgreich etabliert wurde, wartet der virtuelle Drucker auf Ereignisse, die von der SPOOL- Administrator-Task kommen.



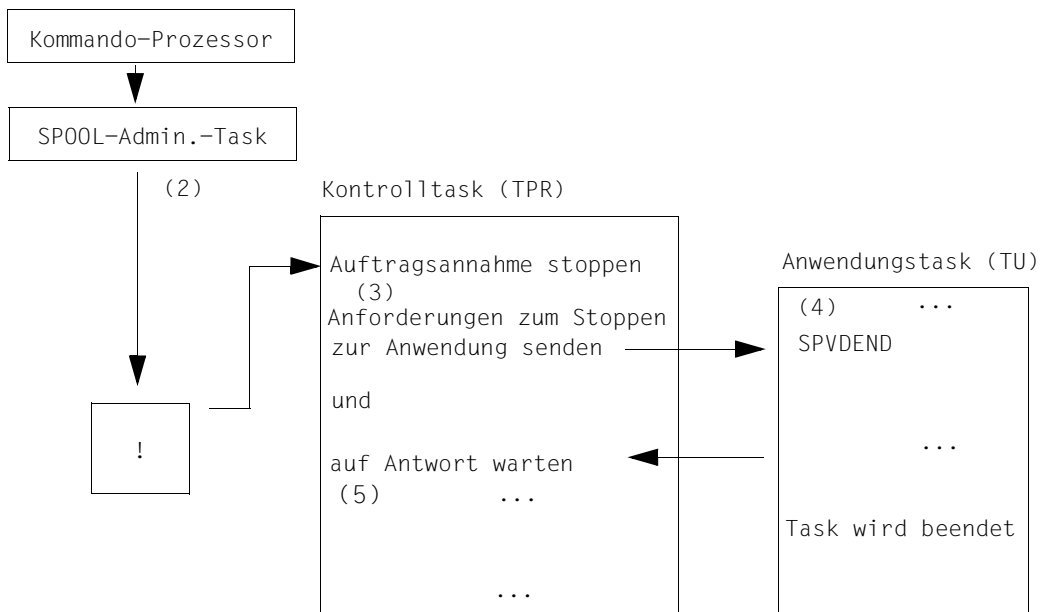
### 6.3.2 Druckaufträge verarbeiten



- (1) Der Benutzer vergibt einen Druckauftrag.
- (2) Gemäß den Verteilungsregeln übergibt die SPOOL-Administrator-Task den Druckauftrag an den virtuellen Drucker.
- (3) Der Druckauftrag wird an die Anwendung übertragen. Danach wartet die Kontrolltask auf die Antwort der Anwendung.
- (4) Die Anwendung erhält die Beschreibung des Druckauftrags (Rückgabe von SPVDRCV), verarbeitet den Druckauftrag und sendet das Ergebnis an den virtuellen Drucker. Die Anwendung gibt Fehlercodes, den bearbeiteten Druckauftrag und dessen neuen Status über die SPVDRET-Schnittstelle zurück an den virtuellen Drucker.
- (5) Der virtuelle Drucker wartet wieder auf Ereignisse, die von der SPOOL-Administrator-Task kommen.
- (6) Die Anwendung setzt sich empfangsbereit und wartet auf Ereignisse, die vom virtuellen Drucker kommen.

### 6.3.3 Verarbeitung stoppen

/STOP-PRINTER-OUTPUT *virdev* (1)



- (1) Der SPOOL-Verwalter gibt das Kommando STOP-PRINTER-OUTPUT.
- (2) Die SPOOL-Administrator-Task informiert die Kontrolltask über das Ende der Verarbeitung.
- (3) Die Kontrolltask überträgt den Stop-Befehl zur Anwendung.
- (4) Die Anwendung beendet ihre Arbeit, gibt die Dialogumgebung mit SPVDEND frei und beendet die Task.
- (5) Der virtuelle Drucker stellt seine Arbeit ein und die Kontrolltask wird beendet.

### 6.3.4 Fehlerbehandlung

1. Fehler beim Initialisieren des virtuellen Druckers

Der virtuelle Drucker wird zwangsweise beendet. An der Konsole wird eine Fehlermeldung ausgegeben.

2. Fehler während der Verbindung mit der Anwendungstask

Sobald der virtuelle Drucker die Anwendungstask erzeugt hat, versucht er, sich mit der Anwendungstask zu verbinden. Nach einer Minute erfolgloser Versuche wird eine Meldung an der Konsole ausgegeben und der virtuelle Drucker gestoppt.

3. Fehler, die bei der Auftragsbearbeitung durch die Anwendung zurückgemeldet werden.

An der Konsole wird eine Fehlermeldung ausgegeben. Der Druckauftrag wird in den Wartezustand versetzt.

### 6.3.5 Wiederaufsetzen nach Fehlern

1. Ausfall des virtuellen Druckers

Wenn der virtuelle Drucker ausfällt, erzeugt der Wiederaufsetzmechanismus des SPOOL eine neue Kontrolltask, die die Verbindung zum Anwendungstask herstellen muss.

2. Probleme mit der Anwendungstask

Wenn die TU-Anwendung ausfällt, beendet die Anwendungstask die Verarbeitung. In diesem Fall informiert der Inter-Task-Kommunikationsmechanismus die Partnertasks vom Ende der TU-Anwendung. Daraufhin wird der virtuelle Drucker beendet.

Sollte die TU-Anwendung nicht mehr auf die Anfragen der Kontrolltask antworten und der virtuelle Drucker auf eine Antwort warten, kann das Kommando STOP-PRINTER-OUTPUT gegeben werden. Damit wird die Anwendungstask beendet und der virtuelle Drucker nach spätestens 10 Minuten gestoppt.

## 6.4 Beispiel: Kommunikation mit einem virtuellen Drucker im synchronen Betrieb

```

*****
*                                                                 *
*      COPYRIGHT (C) SIEMENS AG 1998                             *
*              ALL RIGHTS RESERVED                               *
*                                                                 *
*****
*
* Diese Anwendung kommuniziert mit der Kontrolltask des virtuellen
* Druckers, um Druckaufträge anzufordern.
* Sie prueft den im Kommando /PRINT-DOCUMENT eingegebenen Parameter
* STRING und bearbeitet den Druckauftrag entsprechend dessen
* verschiedenen Werten:
* "CANCEL", der Druckauftrag wird beendet.
* "KEEP", der Druckauftrag wird in die Warteschlange fuer Aufträge
*   im Status KEEP eingereicht.
* "TRANS", dem Druckauftrag wird die Code-Umsetztabelle zugewiesen,
*   deren Name im Operanden STRING folgt. Dazu wird ein MODIFY-
*   PRINT-JOB-ATTRIBUTES ausgefuehrt. Der Parameter VIRTUAL wird
*   dabei auf *NOT-ALLOWED gesetzt, um zu verhindern, dass der
*   Druckauftrag danach wieder dem virtuellen Drucker zugeteilt wird.
* Wenn während der Modifikation ein Fehler auftritt, wird der
* Auftrag in die KEEP-Warteschlange eingehängt, ansonsten in den
* Status WAIT gesetzt.
*
*****

        TITLE 'MACRO DEFINITION'

        MACRO
        SAVE2 &LABEL1
        ST   R10,&LABEL1
        ST   R14,&LABEL1+4
        LR   R10,R15
        MEND

        MACRO
        EXIT &LABEL2
        L    R10,&LABEL2
        L    R14,&LABEL2+4
        @EXIT
        MEND

        TITLE 'VIRTUAL DEVICE  APPLICATION'

VIRTUAL @ENTR TYP=M,TITLE=NO,ENV=SPLSPEC,LOCAL=WORKA

```

```

VIRTUAL AMODE ANY
VIRTUAL RMODE ANY
        @DATA CLASS=B,BASE=R8,DSECT=CONSTANT
        SPACE 1
        L      R8,CONSTADD
*
*      Dialog zwischen Anwendung und virtuellem Drucker
*      initialisieren
*
        @PASS NAME=VIRTINI
        SETUS ON=1
*
*      Anwendung empfangsbereit setzen
*
        @PASS NAME=VIRTRCV
*
*      Dialog zwischen Anwendung und virtuellem Drucker beenden
*
        @PASS NAME=VIRTEND
        SETUS OFF=1
        @EXIT
        @END
CONSTADD DC      A(CONSTANT)

VIRTINI @ENTR TYP=L,TITLE=NO,BASE=R10,LOADSB=YES
        @DATA CLASS=B,BASE=R8,DSECT=CONSTANT
        @DATA CLASS=B,BASE=R13,DSECT=WORKA
        SAVE2 VIRTINIS
*
*      Dialog zwischen Anwendung und virtuellem Drucker
*      initialisieren
*
        MVC APVINI(SPDI#),PLVINI      INIT P/L
        SPVDINI MF=E,PARAM=APVINI
        @IF   NE
        CLC  SPDIMRET,=AL2(SPDIRCOK)  SUCCESSFULLY
        @THEN
        @IF   EQ
        CLC  SPDIMRET,=AL2(SPDIRCER)  DIALOG NOT POSSIBLE
        @THEN
        TERM
        @BEND
        @BEND
*
*      OUTPUT AREA anfordern
*
        REQM 1,16,PARMOD=24
        ST   R1,OUTAREAA

```

```

MVC   OUTAREAL,C4096
@IF   NE
LTR   R15,R15
@THEN
TERM
@BEND
EXIT  VIRTINIS
@END

VIRTEND @ENTR TYP=L,TITLE=NO,BASE=R10,LOADSB=YES
        @DATA CLASS=B,BASE=R8,DSECT=CONSTANT
        @DATA CLASS=B,BASE=R13,DSECT=WORKA
        SAVE2 VIRTENDS
*
*      Dialog zwischen Anwendung und virtuellem Drucker beenden
*
MVC   APVEND(SPVE#),PLVEND      INIT P/L
SPVDEND MF=E,PARAM=APVEND
EXIT  VIRTENDS
@END

VIRTRCV @ENTR TYP=L,TITLE=NO,BASE=R10,LOADSB=YES
        @DATA CLASS=B,BASE=R8,DSECT=CONSTANT
        @DATA CLASS=B,BASE=R13,DSECT=WORKA
        SAVE2 VIRTRCVS
*
*      Empfangsbereit
*
@CYCL
L     R6,OUTAREAA
L     R7,OUTAREAL
MVC   APDRCV(SPVR#),PLDRCV
SPVDRCV MF=M,OUTADDR=(R6),OUTLGT=(R7)
SPVDRCV MF=E,PARAM=APDRCV
@CAS2 SPVRMRET,COMP=CLC
*     CHECK STATUS
@OF   =AL2(SPVRRCOK)
*     WORK RECEIVED
@CAS2 SPVRDAT,COMP=CLI
*     WHICH KIND OF WORK
@OF   SPVRJOB
*     CONTROL TO VIRTUAL PROCESS
@PASS NAME=VIRTDEV
@PASS NAME=VIRTRET
@OF   SPVRSTOP
*     USE=NO OF THE VIRTUAL PRINTER

```

```

EXIT VIRTRCVS
@OFRE
@BEND
@OF  =AL2(SPVRRINIE)
*   INIT CALL EXPECTED
@PASS NAME=VIRTINI
@OF  =AL2(SPVRRREPX)
*   RETRUN EXPECTED
@PASS NAME=VIRTRET
@OF  =AL2(SPVRRTRM)
*   TERMINATION DONE
EXIT VIRTRCVS
@OFRE
@BEND
@BEND
EXIT VIRTRCVS
@END

VIRTRET @ENTR TYP=L,TITLE=NO,BASE=R10,LOADSB=YES
@DATA CLASS=B,BASE=R8,DSECT=CONSTANT
@DATA CLASS=B,BASE=R13,DSECT=WORKA
SAVE2 VIRTRETS
*
*   Druckauftrag und Ergebnis an virtuellen Drucker zurueckgeben
*
SPVDRET MF=E,PARAM=APDRET
EXIT VIRTRETS
@END

VIRTDEV @ENTR TYP=L,TITLE=NO,BASE=R10,LOADSB=YES
@DATA CLASS=B,BASE=R3,DSECT=SCBE
@DATA CLASS=B,BASE=R6,DSECT=STRINGDS
@DATA CLASS=B,BASE=R8,DSECT=CONSTANT
@DATA CLASS=B,BASE=R13,DSECT=WORKA
SAVE2 VIRTDEVS
*
*   Druckauftrag wie gefordert verarbeiten
*
L      R3,SPVROUTA
LH     R6,SCBEVIDS
LA     R6,0(R6,R3)
LR     R2,R6
MVC APDRET(SPVT#),PLDRET      INIT P/L
@CAS2 ACTION,COMP=CLC
*   CANCEL THE JOB

```

```

@OF CAN
SPVDRET MF=M,PACTION=*CANCEL
* KEEP THE JOB
@OF KEEP
SPVDRET MF=M,PACTION=*KEEP
@OF TRANS
* USE THE PRIVATE TRANSLATION TABLE
LA R4,APMOD
LA R6,PLMOD
LA R5,SMA#
LA R7,SMA#
MVCL R4,R6
LR R6,R2
LA R4,TRTNAME
MVI WORKFILE,X'40'
MVC WORKFILE+1(L'WORKFILE-1),WORKFILE
MVC WORKFILE(L'TRTLIB),TRTLIB
LA R5,WORKFILE
LA R2,SCBETSN
MODPJAT MF=M,TRANTAB=((R4),(R5)),
        VIRTUAL=*NOT_ALLOWED,TSN=((R2))
MODPJAT MF=E,PARAM=APMOD
@IF EQ
CLC SMARET,=A(SMAOK)
@THEN
MVC APDRET(SPVT#),PLDRET INIT P/L
SPVDRET MF=M,PACTION=*WAIT
@ELSE
MVC APDRET(SPVT#),PLDRET INIT P/L
SPVDRET MF=M,PACTION=*KEEP
@BEND
@OFRE
@BEND
EXIT VIRTDEVS
@END

TITLE ' STATIC PARAMETER LISTS OF MACROES '
CONSTANT DS OF
OUTAREAA DC A(0)
OUTAREAL DC A(0)
C4096 DC F'4096'
ZERO DC F'0'
REQMRCWO DS F
RCWORK DS CL10
SPTAB DC C'0123456789ABCDEF'
CAN DC CL6'CANCEL'
KEEP DC CL6'KEEP '
MODIFY DC CL6'MODIFY'
TRANS DC CL6'TRANS '
WORKFILE DS CL44

```



```

PLVINI  SPVDINI MF=L          STATIC P/L
PLDRCV  SPVDRCV MF=L         STATIC P/L
PLDRET  SPVDRET MF=L         STATIC P/L
PLVEND  SPVDEND MF=L
PLMOD   MODPJAT MF=L

```

```
TITLE 'VIRTUAL - AUTOMATIC WORK AREA (ADF)'
```

```

*
*   Dummy-Prozedur legt am Ende des Moduls ein ADF an
*

```

```

XXXXXXXX @ENTR TYP=I,ENV=SPLSPEC,TITLE=NO,LOCAL=WORKA
          @END LTORG=NO
VIRTUAL  CSECT ,              BE SURE MAIN CSECT BEFORE @
WORKA    @PAR  D=YES
          PRINT GEN
WORKABEG EQU  *
VIRTINIS DS   2F
VIRTENDS DS   2F
VIRTRCVS DS   2F
VIRTRETS DS   2F
VIRTDEVS DS   2F
APVINI   SPVDINI MF=C

          ORG  APVINI
APDRCV   SPVDRCV MF=C
*
          ORG  APVINI
APDRET   SPVDRET MF=C
*
          ORG  APVINI
APVEND   SPVDEND MF=C
*
          ORG  APVINI
APMOD    MODPJAT MF=C
*
          ORG  APVINI
WORKALEN EQU  *-WORKABEG
WORKA    @PAR  LEND=YES

          SSVSCB D

          SSVSCBD D

STRINGDS DSECT
LENGTHST DS   C
PRGNAME  DS   CL8
STRING   DS   OCL32
ACTION   DS   CL6

```

```
PAR      DS    0CL26
TRTNAME  DS    CL8
TRTLIB   DS    CL18
          ORG   PAR
REFER    DS    CL8
VALUE    DS    CL8
          END   ,
```

---

## 7 Spool-Exits

Ein Exit ist eine Stelle im Betriebssystem oder in einem TU-Programm (TU = Task unprivileged), an der die laufende Verarbeitung durch den Aufruf einer so genannten Exit-Routine unterbrochen werden kann.

Eine Exit-Routine ist eine Komponente, die in der Regel von der BS2000-Systembetreuung erstellt und dem Betriebssystem hinzugefügt wird. Mit dieser Exit-Routine kann eine normalerweise nicht änderbare Betriebssystem-Funktion an spezielle Erfordernisse des RZ-Betriebs angepasst werden. Exit-Routinen können Funktionen zum normalen Systemverhalten hinzufügen oder existierende Funktionen verändern bzw. ersetzen. Sie erscheinen als integrierte Bestandteile des Betriebssystems.

Exit-Routinen können während des Systemlaufs geladen und aktiviert oder deaktiviert, also dynamisch verwaltet werden.

In diesem Kapitel werden die SPOOL-Exits 090, 091, 092, 093, 094, 095, 096 und 097 kurz erklärt. Im Handbuch „[System Exits \(nur in Englisch\)](#)“ finden Sie außer für Exit 093 die Beschreibung dieser Exits und die Auflistung der DSECTs, außerdem eine allgemeine Einführung in die Exits.

## 7.1 Allgemeines zu den Spool-Exits

### Übersicht der Exit-Punkte

Die Exits sind intern durch ihre Exit-Nummer gekennzeichnet. Die folgende Übersicht zeigt die Exit-Punkte und ihren Aufrufzeitpunkt.

Nr.	Exit	Typ/Zeitpunkt des Aufrufs
090	SPOOL-Exit	vor Ausgabe jedes Satzes auf Drucker
091	SPOOL-Deckblatt	vor Ausgabe des Deckblatts
092	SPOOL-Schlussblatt	vor Ausgabe des Schlussblatts
093	Filter-Exit	beim Transfer einer Datei von BS2000 in ein UNIX-System und von einem UNIX-System nach BS2000
094	SPOOL-Ressourcen-Routine	Konvertierung Client- / Server-Ressource
095	SPOOL-Exit	vor der Ausgabe des Kanalprogramms
096	SPOOL-Exit	Server-Auswahl
097	SPOOL-Exit	Steuern der Aufträge für den File-Transfer

Bei Ausgabegeräten kann für jedes Gerät einzeln gesteuert werden, ob SPOOL-Ausgaben auf diesem Gerät durch SPOOL-Exitroutinen bearbeitet werden sollen, siehe Kommando START-PRINTER-OUTPUT.

### Aktuelle SPOOL-Version abfragen

Die SPOOL- (und RSO-)Exit-Routinen müssen die Nummer der aktuellen SPOOL-Version im SCB abfragen:

Exit-Routinen werden unabhängig von den DSSM-Kommandos START-SUBSYSTEM, STOP-SUBSYSTEM, HOLD-SUBSYSTEM und RESUME-SUBSYSTEM geladen, aktiviert und deaktiviert. Aus diesem Grund kann SPOOL entladen und durch eine andere Version ersetzt werden, während die Exit-Routinen aktiviert bleiben und später von der neu geladenen SPOOL-Version aufgerufen werden.

Die Exit-Routinen müssen diese Änderung der SPOOL-Version erkennen und prüfen daher die im SCB hinterlegte Versionsangabe.

Die Versionsnummer ist im Feld SCBESCB der DSECT enthalten, die mit dem Makro SSVSCB erzeugt werden kann.

## Einsatz der SPOOL-Exits

Sind die Exits 90, 91 und 92 gleichzeitig aktiv, entsteht folgende Reihenfolge der Aufrufe:

- EXIT 90: erster Aufruf
- EXIT 91: Routine für Deckblatt
- EXIT 90: Verarbeitungsroutine
- EXIT 92: Routine für Schlussblatt
- EXIT 90: Schlusseruf.

Der erste Aufruf ist bereits vor Dateiöffnung abgearbeitet. Anschließend erfolgt der Aufbau des Deckblattes.

Nach der Erstellung des Schlussblattes erfolgt der Schlusseruf, danach das Schließen der Datei.

Unabhängig davon kann der Exit 95 aktiviert worden sein, um Kanalprogramme zu überwachen, die zu den Druckern gesendet werden.

Bei einer „Family“-Bearbeitung kann entweder nur für die erste Datei oder für jede Datei der „Family“ ein Deckblatt gedruckt werden. Über einen SPSINF-Aufruf (INFO=SPSGEN) in der EXIT-Routine können Sie den GEN-Eintrag lesen und so die entsprechenden Informationen erhalten:

Ist im GEN-Eintrag GENFAM=YES gesetzt, wird für jede Datei des „Family“-PRINTs ein Deckblatt gedruckt, andernfalls nur für die erste Datei (d.h. wenn im SCB das Feld SCBESEQ#=0 gesetzt ist).

Das Layout des Schlussblattes für den lokalen SPOOL finden Sie auf [Seite 611](#).

Mit Hilfe des Exits 093 kann der Inhalt von Dateien angepasst werden, die entweder in einem UNIX-System-basierten CLuster ausgedruckt werden sollen oder von einem UNIX-System-basierten Cluster kommend auf einem BS2000-Drucker ausgedruckt werden sollen.

Mit dem Exit 094 kann die Systembetreuung eine Anwenderoutine erstellen, mit deren Hilfe die Konvertierung von Ressourcen innerhalb einer Client-Server-Umgebung bei Einsatz des Subsystems Distributed Print Services gesteuert wird.

Mit dem Exit 095 kann die Systembetreuung eine Benutzerroutine in die Spool & Print-Services einbinden, die bei jeder Ausgabe des SPOOL in Aktion tritt.

Mit dem Exit 096 kann die Systembetreuung bei Einsatz des Subsystems Distributed Print Services eine Server-Auswahl durch Kombination bestimmter Auswahl-Kriterien vornehmen.

Mit dem Exit 097 kann die Arbeitslast bei der Auftragsvergabe an Dprint DFTM Tasks ausgeglichen werden. Dieser Exit erlaubt eine Kontrolle des Arbeitsflusses zwischen den verschiedenen Dprint-Hosts.

*Hinweis*

Die nachfolgenden Beschreibungen beziehen sich auf das mit SPOOL V3.0 neue Kommando PRINT-DOCUMENT bzw. den neuen Makro PRNTDOC. Die Exit-Funktionen können in gleicher Weise mit dem alten Kommando PRINT-FILE bzw. dem alten Makro PRNT genutzt werden.

## 7.2 SPOOL-OUT-Exit (090)

Diese Exit-Routine wird aufgerufen, nachdem ein Datensatz der auszugebenden Datei gelesen wurde, aber bevor der Datensatz gemäß den Angaben im PRINT-DOCUMENT vom SPOOL weiterverarbeitet wird.

Der Exit ist verwendbar für die Ausgabe auf Zeilendrucker und Laserdrucker (Seitendrucker). Er kann nicht verwendet werden bei Ausgabe auf Magnetband.

Die Exit-Routine kann folgende Maßnahmen des Systems veranlassen:

- den gelesenen Datensatz akzeptieren (verändert oder unverändert)
- den gelesenen Datensatz ersetzen
- den gelesenen Datensatz übergehen
- den gelesenen Datensatz und alle bis Dateiende folgenden Datensätze übergehen

Folgende Informationen werden an die Exit-Routine übergeben:

R0 = Exitnummer "090"  
 R1 = A(EX090-Parameterliste s.u.)  
 R12 = A(TPR Program Manager)  
 R13 = A(Sicherungsbereich)  
 R14 = A(indirekter Rücksprung)  
 R15 = A(Exit-Routine)

Die Returncodes werden im Standardheader eingetragen.

Standard-  
header:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX090 wird im Standardheader ein Returncode übergeben.

aa=Maincode1; bb=Maincode2; cc=Subcode1;  
dd=Subcode2

X'aa'	Erläuterung
00	Gelesenen Datensatz akzeptieren
04	Gelesenen Datensatz durch Inhalt des Ausgabeblockes ersetzen
08	Gelesenen Datensatz übergehen
0C	Gelesenen Datensatz und alle folgenden bis Dateiende übergehen

*Hinweis*

- Die Exit-Routine sollte nicht aktiviert bzw. deaktiviert werden, solange ein SPOOLOUT bearbeitet wird.
- Der Makroaufruf PRNTDOC darf in der Exit-Routine verwendet werden, nicht aber \$PRNT und \$SPRQ.
- Die Exit-Routine erhält die Steuerung bei jedem von der Eingabedatei gelesenen Satz, also auch beim Probedruck der ersten Seite, bis der Operator den Druck startet. Das Gleiche gilt für die RESPOOL-Verarbeitung (HOLD-PRINT-JOB-/RESUME-PRINT-JOB-Kommando), wobei die Exit-Routine die Lage des RESPOOL-Beginns berücksichtigen muss.
- Die notwendigen Informationen zur Verarbeitung von Dateien mit Nicht-Siemens-Steuereichen entnehmen Sie bitte dem PRINT-DOCUMENT-Kommando bzw. dem SPOOL-Control-Block.
- Wenn eigene Ausgabeblocks zur Verfügung gestellt werden, ist eine beliebige Speicherkategorie erlaubt. Es empfiehlt sich aber, den Klasse-5-Speicher zu nehmen. Der Speicher wird vom Exit wieder freigegeben, sobald der Exit die Kontrolle wieder zurückbekommen hat.

**DSECT**

Eine DSECT/CSECT für die Parameterliste (Adresse wird in Register 1 übergeben) kann mit dem Makro EX090 erzeugt werden, wobei "prefix" max. 3 Zeichen lang sein darf.

```
name EX090 D/C ,[prefix]
```

Die Eingabefelder werden nach dem Rücksprung von der rufenden SPOOL-Komponente nicht mehr abgeprüft. Der erste und der letzte Aufruf der Exit-Routine erfolgt, ohne dass ein Satz der Datei bereitgestellt wird (ICLTYP = X'02' bzw. X'03'). Das Byte IPRTYP gibt das Ausgabegerät an.

Die Ausgabefelder werden von der Exit-Routine versorgt. Das Byte IEXRC steuert die von der Exit-Routine gewünschte Funktion. Die Funktionen sind im Anschluss an die DSECT näher beschrieben:



```

SPOOLEX0 EX090 D
1 *
1 *   SPOOL INPUT RECORDS ROUTINE EXIT PARAMETER LIST
1 *
1 SPOOLEX0 IDLKG ID=RC,P=,VER=851,SECT=D,ALIGN=F           753
2      * ,VERSION 851
2 SPOOLEX0 DSECT
1 IBEG      DS      0H
1      FHDR MF=(C,I),EQUATES=NO
2      DS      0A
2 IFHE      DS      OXL8           0   GENERAL PARAMETER AREA HEADER
2 *
2 IIFID     DS      0A           0   INTERFACE IDENTIFIER
2 IFCTU     DS      AL2          0   FUNCTION UNIT NUMBER
2 *
2 *                BIT 15   HEADER FLAG BIT,
2 *                MUST BE RESET UNTIL FURTHER NOTICE
2 *                BIT 14-12 UNUSED, MUST BE RESET
2 *                BIT 11-0  REAL FUNCTION UNIT NUMBER
2 IFCT      DS      AL1          2   FUNCTION NUMBER
2 IFCTV     DS      AL1          3   FUNCTION INTERFACE VERSION NUMBER
2 *
2 IRET      DS      0A           4   GENERAL RETURN CODE
2 ISRET     DS      0AL2         4   SUB RETURN CODE
2 ISR2      DS      AL1          4   SUB RETURN CODE 2
2 ISR1      DS      AL1          5   SUB RETURN CODE 1
2 IMRET     DS      0AL2         6   MAIN RETURN CODE
2 IMR2      DS      AL1          6   MAIN RETURN CODE 2
2 IMR1      DS      AL1          7   MAIN RETURN CODE 1
2 IFHL      EQU     8            8   GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *
1 *   EQUATES FOR EXIT RETURN INFORMATION
1 *
1 IACCP     EQU     0            ACCEPT RECORD AS IT IS
1 IADD      EQU     4            ADD RECORD(S)
1 IDEL      EQU     8            DELETE (SKIP) RECORD
1 IEOF      EQU     12           TERMINATE PRINT (SIMULATE EOF)
1 *   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
1 *
1 IREC@     DS      A            INPUT RECORD ADDRESS
1 ISCB@     DS      A            SCB ADDRESS
1 IRCLLEN   DS      Y            INPUT RECORD LENGTH
1 IPRTYP    DS      AL1         TYPE OF PRINT
1 ILINES    EQU     32           LINES PRINTER TYPE           751
1 IPAGES    EQU     33           PAGE PRINTER TYPE           751
1 ILOADL    EQU     34           LOADABLE LINE PRINTER TYPE     751
1 IRBP      EQU     35           RBP PRINTER TYPE              751
1 IRSOPB    EQU     36           PUBLIC RSO PRINTER           752
1 IRSOPR    EQU     37           PRIVATE RSO PRINTER           752
1 IAPAPR    EQU     38           APA PRINTERS

```

1	IPUNCH	EQU	48	PUNCH TYPE	751
1	ICLTYP	DS	AL1	CALL TYPE	
1	ICLREC	EQU	1	RECORD CALL TYPE	
1	ICLFST	EQU	2	NO RECORD FIRST TIME CALL	
1	ICLLST	EQU	3	NO RECORD LAST TIME CALL	
1		DS	CL4	RESERVED FOR EXTENSION	
1	*				
1	*			OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE	
1	*				
1	IBLK@	DS	A	DATA BLOCK ADDRESS	
1	IBLKLE	DS	Y	DATA BLOCK LENGTH	
1	IEXRC	DS	AL1	R-C FROM EXIT ROUTINE MAY BE	
1	*			STORED HERE IN ADDITION TO	
1	*			THE MAIN RETURN CODE	
1	IBEXRC	DS	AL1	BASE R-C FROM EXIT ROUTINE	
1	*			MAY BE STORED HERE	
1	IBRCOK	EQU	0	BASE R-C O.K.	
1	INEXS	EQU	4	EXIT ROUTINE NOT EXISTENT	
1	INACT	EQU	8	EXIT ROUTINE NOT ACTIVE	
1	IPARER	EQU	12	PARAMETER ERROR	
1		DS	CL8	RESERVED FOR EXTENSION	
1	*				
1	*			INPUT FIELDS FOR THE INPUT EXIT ROUTINE	
1	*				
1	IPIR	DS	A	PRINTER INFORMATION RECORD ADDR.	755
1	IDVTYP	DS	AL1	DEVICE TYPE	755
1		DS	CL11	RESERVED FOR EXTENTION	755
1	ILEN	EQU	*-SPOOLEXO	ROUTINE EXIT P/L LENGTH	

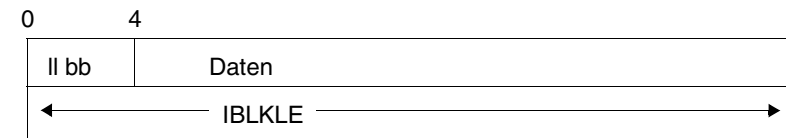
## Funktionen der Exit-Routine

*Returncode X'00': gelesenen Satz akzeptieren*

Die Adresse des Satzes und die Satzlänge werden in den Feldern IREC@ bzw. IRCLEN an die Exit-Routine übergeben. Der Satz enthält kein Satzlängenfeld.

Die Exit-Routine kann den Satz unverändert vom SPOOL akzeptieren lassen oder ihn ändern. Wenn der Satz dabei nicht länger wird, kann er direkt im Eingabepuffer verändert werden. Die Felder IBLK@ und IBLKLE müssen in diesem Fall mit der Adresse des Satzes aus IREC@ und der gültigen Satzlänge versorgt werden. Wird der Satz durch die Änderung verlängert, muss die Exit-Routine den Satz in einem eigenen, über IBLK@ adressierten Puffer bereitstellen.

Aufbau des Puffers



IBLK@

ll Datenlänge + 4 (IBLKLE)  
bb Leerzeichen

*Returncode X'04': gelesenen Satz ersetzen*

SPOOL ersetzt den eingelesenen Satz durch die Sätze, die von der Exit-Routine im Ausgabeblock (Adresse in IBLK@) bereitgestellt wurden:

ll bb	Daten 1	ll bb	Daten 2
	ll bb	Daten 3	
ll bb	Daten 4	. . .	ll bb
Daten n-1	ll bb	Daten n	

Die Exit-Routine muss die Felder IBLK@ und IBLKLE versorgen. Soll der gelesene Satz Bestandteil des Ausgabeblocks sein, muss ihn die Exit-Routine dorthin übertragen.

*Returncode X'08': gelesenen Satz übergehen*

Die Exit-Routine muss nur den Returncode setzen, weitere Aktionen sind nicht erforderlich.

*Returncode X'0C': alle Sätze bis Dateiende übergehen*

Die Exit-Routine muss nur den Returncode setzen, weitere Aktionen sind nicht erforderlich.

### Operanden im PRINT-DOCUMENT-Kommando/PRNTDOC-Makro

Die Exit-Routine muss alle in PRINT-DOCUMENT-Kommandos angegebenen Operanden beachten! Die entsprechenden Werte können dem SPOOL-Kontrollblock entnommen werden.

*RECORD-PART=\*PARAMETERS(FIRST-CHARACTER = n):* Alle von der Exit-Routine eingefügten Sätze müssen länger als n Bytes sein, sonst werden sie nicht gedruckt.

*OUTPUT-PART=\*RANGE(FROM = +n):*

Die Sätze 1–n werden an die Exit-Routine übergeben.

*OUTPUT-PART=\*LAST(LAST = -n):* Die Exit-Routine erhält die Steuerung zunächst für alle Sätze von 1–m (Dateiende, keine Ausgabe). Danach setzt SPOOL den FROM-Wert auf z = (m–n+1) und arbeitet die Datei ab wie bei FROM = +z.

*ADDITIONAL-COPIES = n:* die Exit-Routine kann bei wiederholter Ausgabe einer Datei nicht zwischen den einzelnen Kopien unterscheiden, d.h. es gibt keinen Unterschied in den übergebenen Parametern oder Datensätzen.

### Beispiel für den Exit 090

```
MACRO          860          941109   32076095 SPOOL          U
&L          EX090 &D,&P
.*****
.*
.*          COPYRIGHT (C) SIEMENS AG 1990          *
.*          ALL RIGHTS RESERVED          *
.*
.*          LCLC &NAME
&NAME        SETC  '&P'
*
*          SPOOL INPUT RECORDS ROUTINE EXIT PARAMETER LIST
*
          AIF  ('&P' NE '') .L5
&P          SETC  'I'
.L5         ANOP
          AIF  ('&P' NE '*') .L6
&P          SETC  ''
```

```

.L6      ANOP
        AIF  (&L NE  ' ').L01
&L      SETC  '&P.RC'
.L01    ANOP
&L      MFCHK MF=&D,DMACID=E90,ALIGN=F,SUPPORT=(C,D),ENTRY=NO,      *
        PREFIX=I,MACID=E90,DNAME=C
*
*   VERSION 850 : ADAPTATION V3 : STANDARD HEADER FOR RETURN INFO.
*
*   STANDARD HEADER RETURN INFORMATION
*
&P.BEG  DS    OH
        FHDR MF=(C,&P),EQUATES=NO
*
*   EQUATES FOR EXIT RETURN INFORMATION
*
&P.ACCEPT EQU  0          ACCEPT RECORD AS IT IS
&P.ADD      EQU  4          ADD RECORD(S)
&P.DEL      EQU  8          DELETE (SKIP) RECORD
&P.EOF      EQU 12          TERMINATE PRINT (SIMULATE EOF)
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
&P.REC@    DS    A          INPUT RECORD ADDRESS
&P.SCB@    DS    A          SCB ADDRESS
&P.RCLEN   DS    Y          INPUT RECORD LENGTH
&P.PRTP   DS    AL1        TYPE OF PRINT
&P.LINES   EQU  32          LINES PRINTER TYPE          751
&P.PAGES   EQU  33          PAGE PRINTER TYPE          751
&P.LOADL   EQU  34          LOADABLE LINE PRINTER TYPE  751
&P.RBP     EQU  35          RBP PRINTER TYPE          751
&P.RSOPB   EQU  36          PUBLIC RSO PRINTER        752
&P.RSOPR   EQU  37          PRIVATE RSO PRINTER       752
&P.APAPR   EQU  38          APA PRINTERS
&P.PUNCH   EQU  48          PUNCH TYPE                751
&P.CLTP   DS    AL1        CALL TYPE
&P.CLREC   EQU  1          RECORD CALL TYPE
&P.CLFST   EQU  2          NO RECORD FIRST TIME CALL
&P.CLLST   EQU  3          NO RECORD LAST TIME CALL
        DS    CL4          RESERVED FOR EXTENSION
*
*   OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
*
&P.BLK@    DS    A          DATA BLOCK ADDRESS
&P.BLKLE   DS    Y          DATA BLOCK LENGTH
&P.EXRC    DS    AL1        R-C FROM EXIT ROUTINE MAY BE
*
*
*
&P.BEXRC   DS    AL1        BASE R-C FROM EXIT ROUTINE

```

```

*
&P.BRCOK EQU 0          MAY BE STORED HERE
&P.NEXS EQU 4          BASE R-C O.K.
&P.NACT EQU 8          EXIT ROUTINE NOT EXISTENT
&P.PARER EQU 12        EXIT ROUTINE NOT ACTIVE
                DS CL8  PARAMETER ERROR
                CL8    RESERVED FOR EXTENSION
*
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
&P.PIR DS A            PRINTER INFORMATION RECORD ADDR. 755
&P.DVTYP DS AL1        DEVICE TYPE 755
                DS CL11  RESERVED FOR EXTENTION 755
                AIF ('&L' EQ '').NOLAB
&P.LEN EQU *-&L        ROUTINE EXIT P/L LENGTH
                MEXIT
.NOLAB ANOP
                AIF ('&NAME' NE '*').NOLAB1
&NAME SETC ''
.NOLAB1 ANOP
&P.LEN EQU *-&P.BEG    ROUTINE EXIT P/L LENGTH
                MEND

```

## 7.3 Exits für Deckblatt und Schlussblatt (091/092)

Mit diesen Exits wird die Gestaltung der Deckblätter / Schlussblätter bei Ausgaben auf Drucker durch Exit-Routinen gesteuert.

Die Exit-Routinen können dabei die Standardausgabe des SPOOL-Systems unterdrücken, durch eine Alternativausgabe ersetzen oder zusätzlich zur Standardausgabe eine Alternativausgabe veranlassen.

Alternativausgaben für Deck- oder Schlussblätter werden von der Systembetreuung zur Verfügung gestellt und vom Anwender mit den Operanden HEADER-EXIT-NUMBER und TRAILER-EXIT-NUMBER im Kommando PRINT-DOCUMENT angefordert. Die Standardausgabe kann unabhängig von einer EXIT-Routine durch einen Text (Operand HEADER-PAGE-TEXT im Kommando PRINT-DOCUMENT) modifiziert werden.

### *Beispiel*

Festgelegt werden muss ein Standardname, z.B.

HEADERPAGETEXT.n,

der eine SAM-, ISAM-Datei oder eine Jobvariable bezeichnet. Darin können Sie Texte hinterlegen, die Sie mit dem Kommando

```
PRINT-DOCUMENT . . . , LAYOUT-CONTROL=*PARAMETERS(COVER-PAGES= -
  *PARAMETERS(HEADER-EXIT-NUMBER= . . . ))
```

ansprechen können.

Die Exit-Routine muss auf die Datei

\$userid.HEADERPAGETEXT.n

zugreifen und den Text in den Puffer für das Deckblatt eintragen. Eine Auswahl erfolgt erst, wenn die Exit-Routine aufgerufen ist. Diese Exit-Routinen sichern und stellen alle Register wieder her.

Die Routinen werden aufgerufen:

- wenn kein Deck- bzw. Schlussblatt erzeugt wurde
- bei bestehendem Deck- bzw. Schlussblatt, um den Verwaltungspuffer wiederaufzubauen

**Deckblatt-Exit (091)**

Folgende Informationen werden an die EXIT-Routine übergeben:

- R0 = Exitnummer "091"
- R1 = A(EX091-Parameterliste s.u.)
- R12 = A(TPR Program Manager)
- R13 = A(Sicherungsbereich)
- R14 = A(indirekter Rücksprung)
- R15 = A(Exit-Routine)

Die Returncodes werden im Standardheader eingetragen.

Standard-  
header:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX091 und EX092 wird im Standardheader ein Returncode übergeben.  
aa=Maincode1; bb=Maincode2; cc=Subcode1;  
dd=Subcode2

<b>X'aa'</b>	<b>Erläuterung</b>
00	Kein Deck- bzw. Schlussblatt
01	Ohne Änderung Systemdeck- bzw. -schlussblatt drucken
02	Angegebenes alternatives Deck- bzw. Schlussblatt drucken
03	Das Systemdeck- bzw. Schlussblatt ohne Änderung und das angegebene alternative Deck- bzw. Schlussblatt drucken
04	Nur für Exit 92. Kein Schlussblatt. Der letzte Vorschub auf den Vertikalabulator "Kanal 1" soll nicht durchgeführt werden



**DSECT**

Eine DSECT/CSECT für die Parameterliste (Adresse wird in Register 1 übergeben) kann mit dem Makro EX091 erzeugt werden, wobei "prefix" max. 3 Zeichen lang sein darf.

name EX091 D/C ,[prefix]

```

SPOOLEX1 EX091 D
1 *
1 *   SPOOL HEADER   PAGE   ROUTINE EXIT PARAMETER LIST
1 *
1 SPOOLEX1 IDLKG ID=HD,P=,VER=851,SECT=D,ALIGN=F           754
2      *,VERSION 851
2 SPOOLEX1 DSECT
1 IBEG      DS      0H
1          FHDR MF=(C,I),EQUATES=NO
2          DS      0A
2 IFHE      DS      0XL8           0   GENERAL PARAMETER AREA HEADER
2 *
2 IIFID     DS      0A           0   INTERFACE IDENTIFIER
2 IFCTU     DS      AL2          0   FUNCTION UNIT NUMBER
2 *
2 *                BIT 15   HEADER FLAG BIT,
2 *                MUST BE RESET UNTIL FURTHER NOTICE
2 *                BIT 14-12 UNUSED, MUST BE RESET
2 *                BIT 11-0   REAL FUNCTION UNIT NUMBER
2 IFCT      DS      AL1          2   FUNCTION NUMBER
2 IFCTV     DS      AL1          3   FUNCTION INTERFACE VERSION NUMBER
2 *
2 IRET      DS      0A           4   GENERAL RETURN CODE
2 ISRET     DS      0AL2         4   SUB RETURN CODE
2 ISR2      DS      AL1          4   SUB RETURN CODE 2
2 ISR1      DS      AL1          5   SUB RETURN CODE 1
2 IMRET     DS      0AL2         6   MAIN RETURN CODE
2 IMR2      DS      AL1          6   MAIN RETURN CODE 2
2 IMR1      DS      AL1          7   MAIN RETURN CODE 1
2 IFHL      EQU     8           8   GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *
1 *   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
1 *
1 IEXNH     EQU     0           NO HEADER PAGE OUTPUT
1 IEXSY     EQU     1           SYSTEM HEADER UNALTERED
1 IEXAL     EQU     2           ALTERNATIVE HEADER
1 IEXBO     EQU     3           SYSTEM AND ALTERNATIVE HEADER
1 *   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
1 *
1 IPRTYP    DS      C           PRINTER TYPE
1 ILINES    EQU     X'20'       LINES PRINTER TYPE
1 IPAGES    EQU     X'21'       PAGE PRINTER TYPE

```

753

1	ILOADL	EQU	X'22'	LOADABLE LINE PRINTER TYPE	753
1	IRBP	EQU	X'23'	RBP PRINTER TYPE	753
1	IRSOPB	EQU	X'24'	PUBLIC RSO PRINTER	753
1	IRSOPR	EQU	X'25'	PRIVATE RSO PRINTER	753
1	IAPAPR	EQU	X'26'	APA PRINTER	
1	IPUNCH	EQU	X'30'	PUNCH TYPE	753
1	ICLTYP	DS	C	CALL TYPE	
1	INORM	EQU	X'01'	NORMAL CALL	754
1	IRETRY	EQU	X'02'	RETRY	754
1	INOBUF	EQU	X'04'	NO BUFFER	754
1	ISMLHD	EQU	X'08'	SMALL HEADER	802
1		DS	CL2	RESERVED	
1	ISCB@	DS	A	SCB ADDRESS	
1	ISYIB@	DS	A	SYSTEM BUFFER	
1	ILNPG	DS	F	# OF LINES PER PAGE	
1	ICHLN	DS	F	# OF CHAR. PER LINE	
1	*				
1	*			OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE	
1	*				
1	IUSOB@	DS	A	USER BUFFER ADDRESS	
1	IUSOBL	DS	F	USER BUFFER LENGTH	
1	ISKIP#	DS	F	# SKIP RECORDS	
1	ISYDF	DS	F	SYSTEM DUPL FACT	
1	IUSDF	DS	F	USER DUPL FACT	
1		DS	CL12	RESERVED FOR EXTENSION	
1	*				
1	*			INPUT FIELDS FOR THE INPUT EXIT ROUTINE	
1	*				
1	IPIR	DS	A	PRINTER INFORMATION RECORD ADDR.	756
1	IDVTYP	DS	AL1	DEVICE TYPE	756
1		DS	CL11	RESERVED FOR EXTENTION	756
1	ILEN	EQU	*-SPOOLEX1	ROUTINE EXIT P/L LENGTH	

Vor Aufruf der Exit-Routine versorgte Felder:

IPRTYP	Ausgabegerät
	ILINES      Zeilendrucker
	IPAGES      Laserdrucker / Seitendrucker
	ILOADL      Drucker mit ladbarem VFB
	IRSOPB      gemeinschaftlicher RSO-Drucker
	IRSOPR      privater RSO-Drucker
	IAPAPR      APA-Drucker
ICLTYP	Art des Aufrufs
	INORM      normaler Aufruf
	IRETRY      wiederholter Aufruf: wenn der Speicherbereich für das Alternativdeckblatt zerstört oder nicht korrekt aufgebaut ist
	INOBUF      kein System-Standarddeckblatt: es wird keine Pufferadresse ISYIB@ übergeben; die Exit-Routine kann eine Alternativausgabe veranlassen, entsprechend ILNPG und ICHLN (s.u.)
	FSMLHD      kurzes Deckblatt (nach einer Jobunterbrechung)
ISCB@	Adresse des SPOOL-Kontrollblocks (SCB)
ISYIB@	Adresse des Systempuffers
ILNPG	Anzahl Zeilen pro Deckblatt-Seite
ICHLN	Anzahl Zeichen pro Deckblatt-Zeile

Felder, die von der Exit-Routine versorgt werden können:

IUSOB@	Klasse-3-Speicheradresse, an der das Alternativ-Deckblatt bereitgestellt ist
IUSOBL	Länge des Puffers für das Alternativdeckblatt, entsprechend dem von der Exit-Routine mit \$GETMEM angeforderten Pufferbereich
ISKIP#	Anzahl Sätze des Alternativdeckblatts, die bei einer Wiederholung übergangen, d.h. nicht gedruckt werden sollen
ISYDF	Wiederholungsfaktor für die Ausgabe des Standarddeckblatts; Standard: 1
IUSDF	Wiederholungsfaktor für die Ausgabe des Alternativdeckblatts; Standard: 1
IPIR	Adresse des speziellen Drucker-Informationen-Satzes. Für genauere Informationen wird die Hardware-Beschreibung benötigt. (Unterstützung zur Zeit nur für LP65-Drucker)
IDVTYP	Gerätetyp Genauere Informationen dazu siehe Makro SPSINF (auf <a href="#">Seite 199</a> ).

**Schlussblatt-Exit (092)**

Folgende Informationen werden an die Exit-Routine übergeben:

R0 = Exitnummer "092"  
 R1 = A(EX092-Parameterliste s.u.)  
 R12 = A(TPR Program Manager)  
 R13 = A(Sicherungsbereich)  
 R14 = A(indirekter Rücksprung)  
 R15 = A(Exit-Routine)

**DSECT**

Eine DSECT/CSECT für die Parameterliste (Adresse in R1) kann mit dem Makro EX092 erzeugt werden, wobei "prefix" max. 3 Zeichen lang sein darf.

name EX092 D/C ,[prefix]

```

      SPOOLEX2 EX092 D
1 *
1 *   SPOOL TRAILER PAGE   ROUTINE EXIT PARAMETER LIST
1 *
1 SPOOLEX2 IDLKG ID=TR,P=,VER=851,SECT=D,ALIGN=F              754
2      *,VERSION 851
2 SPOOLEX2 DSECT
1 IBEG      DS      0H
1          FHDR    MF=(C,I),EQUATES=NO
2          DS      0A
2 IFHE      DS      0XL8          0   GENERAL PARAMETER AREA HEADER
2 *
2 IFID      DS      0A          0   INTERFACE IDENTIFIER
2 IFCTU     DS      AL2          0   FUNCTION UNIT NUMBER
2 *
2 *          BIT 15   HEADER FLAG BIT,
2 *          MUST BE RESET UNTIL FURTHER NOTICE
2 *          BIT 14-12 UNUSED, MUST BE RESET
2 *          BIT 11-0  REAL FUNCTION UNIT NUMBER
2 IFCT      DS      AL1          2   FUNCTION NUMBER
2 IFCTV     DS      AL1          3   FUNCTION INTERFACE VERSION NUMBER
2 *
2 IRET      DS      0A          4   GENERAL RETURN CODE
2 ISRET     DS      0AL2         4   SUB RETURN CODE
2 ISR2      DS      AL1          4   SUB RETURN CODE 2
2 ISR1      DS      AL1          5   SUB RETURN CODE 1
2 IMRET     DS      0AL2         6   MAIN RETURN CODE
2 IMR2      DS      AL1          6   MAIN RETURN CODE 2
2 IMR1      DS      AL1          7   MAIN RETURN CODE 1
2 IFHL      EQU     8           8   GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *
1 *   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1

```

```

1 *
1 IEXNT   EQU   0           NO TRAIL. PAGE OUTPUT
1 IEXSY   EQU   1           SYSTEM TRAIL. UNALTERED
1 IEXAL   EQU   2           ALTERNATIVE TRAILER PAGE
1 IEXBO   EQU   3           SYSTEM AND ALTERNATIVE TRAILER
1 *      INPUT FIELDS FOR THE INPUT EXIT ROUTINE
1 *
1 IPRTYP  DS     C           PRINTER TYPE
1 ILINES  EQU   X'20'       LINES PRINTER TYPE
1 IPAGES  EQU   X'21'       PAGE PRINTER TYPE           753
1 ILOADL  EQU   X'22'       LOADABLE LINE PRINTER TYPE  753
1 IRBP    EQU   X'23'       RBP PRINTER TYPE           753
1 IRSOPB  EQU   X'24'       PUBLIC RSO PRINTER         753
1 IRSOPR  EQU   X'25'       PRIVATE RSO PRINTER        753
1 IAPAPR  EQU   X'26'       APA PRINTERS
1 IPUNCH  EQU   X'30'       PUNCH TYPE                 753
1 ICLTYP  DS     C           CALL TYPE
1 INORM   EQU   X'01'       NORMAL CALL                754
1 IRETRY  EQU   X'02'       RETRY                      754
1 INOBUF  EQU   X'04'       NO BUFFER                  754
1         DS     CL2       RESERVED
1 ISCB@   DS     A           SCB ADDRESS
1 ISYIB@  DS     A           SYSTEM BUFFER
1 ILNPG   DS     F           # OF LINES PER PAGE
1 ICHLN   DS     F           # OF CHAR. PER LINE
1 *
1 *      OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
1 *
1 IUSOB@  DS     A           USER BUFFER ADDRESS
1 IUSOBL  DS     F           USER BUFFER LENGTH
1 ISKIP#  DS     F           # SKIP RECORDS
1 ISYDF   DS     F           SYSTEM DUPL FACT
1 IUSDF   DS     F           USER DUPL FACT
1         DS     CL12      RESERVED FOR EXTENSION
1 *
1 *      INPUT FIELDS FOR THE INPUT EXIT ROUTINE
1 *
1 IPIR    DS     A           PRINTER INFORMATION RECORD ADDR. 756
1 IDVTYP  DS     AL1        DEVICE TYPE                 756
1         DS     CL11      RESERVED FOR EXTENTION      756
1 ILEN    EQU   *-SPOOLEX2  ROUTINE EXIT P/L LENGTH

```

Vor Aufruf der Exit-Routine versorgte Felder:

YDDXMAIN	Indikator für das Hauptereignis YDDXAPPL Hauptereignis 'APP' (betr. Anwendungen)
IPRTYP	Ausgabegerät ILINES Zeilendrucker IPAGES Laserdrucker / Seitendrucker ILOADL Drucker mit ladbarem VFB IRSOPB gemeinschaftlicher RSO-Drucker IRSOPR privater RSO-Drucker IAPAPR APA-Drucker
ICLTYP	Art des Aufrufs INORM normaler Aufruf IRETRY wiederholter Aufruf: wenn der Speicherbereich für das Alternativschlussblatt zerstört oder nicht korrekt aufgebaut ist INOBUF kein System-Standardschlussblatt: es wird keine Pufferadresse ISYIB@ übergeben; die Exit-Routine kann eine Alternativausgabe veranlassen, entsprechend ILNPG und ICHLN (s.u.) FSMLHD kurzes Schlussblatt (nach einer Jobunterbrechung)
ISCB@	Adresse des SPOOL-Kontrollblocks (SCB)
ISYIB@	Adresse des Systempuffers
ILNPG	Anzahl Zeilen pro Schlussblatt-Seite
ICHLN	Anzahl Zeichen pro Schlussblatt-Zeile

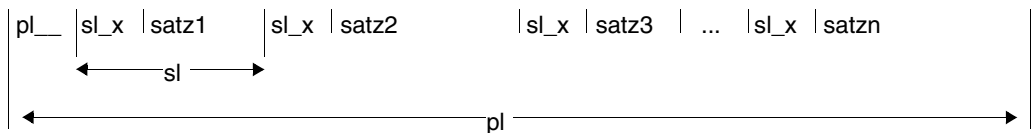
Felder, die von der Exit-Routine versorgt werden können:

IUSOB@	Klasse-3-Speicheradresse, an der das Alternativ-Schlussblatt bereitgestellt ist
IUSOBL	Länge des Puffers für das Alternativschlussblatt, entsprechend dem von der Exit-Routine mit \$GETMEM angeforderten Pufferbereich
ISKIP#	Anzahl Sätze des Alternativschlussblatts, die bei einer Wiederholung übergangen, d.h. nicht gedruckt werden sollen
ISYDF	Wiederholungsfaktor für die Ausgabe des Standardschlussblatts; Standard: 1
IUSDF	Wiederholungsfaktor für die Ausgabe des Alternativschlussblatts; Standard: 1
IPIR	Adresse des speziellen Drucker-Informations-Satzes. Für genauere Informationen wird die Hardware-Beschreibung benötigt. (Unterstützung zur Zeit nur für LP65-Drucker)
IDVTYP	Gerätetyp Genauere Informationen dazu siehe SPOOL-Makro SPSINF (auf <a href="#">Seite 199</a> ).

## Aufbau der Puffer für Deck-/Schlussblätter

### Alternativ-Ausgabe

Für den Aufbau von Alternativ-Ausgaben muss die Exit-Routine einen Puffer im Klasse-3-Speicher reservieren. Die Deck-/Schlussblätter werden aus Sätzen variabler Länge aufgebaut, d.h. mit vorangestelltem 4-Byte-Satzlängenfeld. Die ersten 4 Bytes des Puffer enthalten linksbündig die genutzte Pufferlänge, höchstens 32767 Bytes = X'7FFF'.



- pl genutzte Pufferlänge  
 sl Satzlänge/Zeilenlänge + 4  
 \_ Leerzeichen (X' 40' )  
 x Kennzeichen für Ausgabotyp (nur SPOOL-intern ausgewertet)  
 X' 0A' Drucker
- satz bei PRINT-Ausgabe: 1 Byte Drucksteuerzeichen (s. Tabelle) + Text
- | Steuerzeichen | Bedeutung                         |
|---------------|-----------------------------------|
| X' 40'        | Kein Vorschub vor dem Drucken     |
| X' 41'        | 1 Zeile Vorschub vor dem Drucken  |
| X' 42'        | 2 Zeilen Vorschub vor dem Drucken |
| X' 43'        | 3 Zeilen Vorschub vor dem Drucken |
- Andere Steuerzeichen können zu unerwünschten Ergebnissen führen, da sie nicht von jedem Gerätetyp verstanden werden.

## Systemdeckblatt

Das Deckblattformat ist abhängig von Papiergröße und verwendetem Zeichensatz. Wie viele Zeilen auf eine Druckseite ausgegeben werden können, lässt sich folgendermaßen bestimmen:

$$\text{Max. Zeilenzahl} = \text{Papierformat in Zoll} * \text{Zeilen / Zoll}$$

Ausnahme: Wird bei Laserdruckern das Deckblatt unter LOOP-Steuerung ausgedruckt, ist die Zeilenzahl durch den VFB vorgegeben!

Die Zeilen des Deckblatts werden entsprechend der unten beschriebenen Rangfolge gedruckt (abhängig von der maximalen Zeilenzahl).

### Rangfolge

Adress- und Identifikationsfeld (24 Zeilen)	} in "Riesen"-Lettern
Benutzertext (10 Zeilen) + 2 Leerzeilen	
Auftragsname (10 Zeilen) + 2 Leerzeilen	
Benutzerkennung (10 Zeilen) + 2 Leerzeilen	
Abrechnungsnummer (10 Zeilen) + 2 Leerzeilen	

Stehen weniger als 24 Zeilen zur Verfügung, wird kein Standard-Deckblatt gedruckt. Die Exit-Routine kann jedoch geeignete Maßnahmen treffen.

Jede gedruckte Zeile wird als Satz variabler Länge gespeichert, d.h. mit einem vorangestellten Satzlängelfeld (4 Bytes). Leerzeilen enthalten ein Leerzeichen (X'40') als Satzhalt.



*Beispiel: Aufbau eines Standard-Deckblatts, abhängig von Zeichen- und Zeilendichte*

Format (Seitengröße): 9 x 11 Zoll

Zeichensatz A: 15 Zeichen/Zoll; 8 Zeilen/Zoll (= 15 cpi / 8 lpi)

Zeichensatz B: 12 Zeichen/Zoll; 6 Zeilen/Zoll (= 12 cpi / 6 lpi)

	Zeichensatz A (15 cpi / 8 lpi)	Zeichensatz B (12 cpi / 6 lpi)
Anzahl Zeichen pro Zeile	11 * 15 = 165	11 * 12 = 132
Anzahl Zeilen pro Seite	9 * 8 = 72	9 * 6 = 54
<b>Deckblatt-Aufbau</b>		
Adressfeld (24 Zeilen)	freie Zeilen: 48	freie Zeilen: 30
Auftragsname (10 + 2 Zeilen)	36	18
Benutzerkennung (10 + 2 Zeilen)	24	6
Abrechnungsnummer (10 + 2 Zeilen)	12	-
Bemerkung	Deckblatt wird vollständig ausgedruckt	Abrechnungsnummer fehlt
Länge der Sternchenzeilen	165 <sup>1*)</sup>	132 <sup>1*)</sup>

## Papiervorschub

Nach Drucken des Systemschlussblatts wird das Folgeblatt auf die erste Zeile positioniert. Das gilt auch, wenn kein Deck- oder Schlussblatt gedruckt wird.

## Anwendungshinweise

Die sinnvolle Verwendung der oben beschriebenen Exits setzt geeignete Vereinbarungen zwischen Systembetreuung und Kunden voraus, z.B. einen Standardnamen für eine SAM- oder ISAM-Datei (oder auch eine Jobvariable). In dieser Datei kann sowohl der privilegierte als auch der nichtprivilegierte Benutzer ein Alternativ-Deckblatt aufbauen.

Beispiel für den Aufbau eines Standardnamens:

HEADERPAGETEXT.n,  $0 \leq n \leq 2147.483.639$

Das Alternativdeckblatt kann dann ausgedruckt werden mit PRINT-DOCUMENT-Kommando (Operand HEADER-EXIT-NUMBER = n) oder PRNTDOC-Makro (Operand COVPAGE).

Die Exit-Routine muss dann auf die Datei \$userid.HEADERPAGETEXT.n zugreifen und den Deckblatt-Puffer mit dem Alternativtext versorgen.

Ob ein Deck-/Schlussblatt gedruckt wird, hängt ab von der Anzahl der Zeichen, die pro Zeile gedruckt werden können.

Wenn bei den Exits 091/092 eigene Ausgabeblöcke zur Verfügung gestellt werden, ist der Klasse-3-Speicher erforderlich. Der Speicher wird von SPOOL, nicht von den Exits wieder freigegeben.

### Beispiel für den Exit 091

```

MACRO          865          960110   32733508 SPOOL          U
&L          EX091 &D,&P
.*****
.*
.*          COPYRIGHT (C) SIEMENS AG 1990          *
.*          ALL RIGHTS RESERVED          *
.*          *          *
.*          *****
          LCLC &NAME
&NAME       SETC  '&P'
*
*   SPOOL HEADER PAGE ROUTINE EXIT PARAMETER LIST
*
          AIF  ('&P' NE '').L5
&P         SETC  'I'
.L5        ANOP  ,
          AIF  ('&P' NE '*').L6
&P         SETC  ''
.L6        ANOP  ,
          AIF  (&L NE '').L01
&L         SETC  '&P.HD'
.L01       ANOP  ,
&L         MFCHK MF=&D,DMACID=E91,ALIGN=F,SUPPORT=(C,D),ENTRY=NO,          *
          PREFIX=I,MACID=E91,DNAME=C
.*
.*   VERSION 850 : ADAPTATION V3 : STANDARD HEADER FOR RETURN INFO.
*
*   STANDARD HEADER RETURN INFORMATION
*
&P.BEG     DS     OH
          FHDR  MF=(C,&P),EQUATES=NO
*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
*
&P.EXNH    EQU    0          NO HEADER PAGE OUTPUT
&P.EXSY    EQU    1          SYSTEM HEADER UNALTERED

```

```

&P.EXAL EQU 2 ALTERNATIVE HEADER
&P.EXBO EQU 3 SYSTEM AND ALTERNATIVE HEADER
*
* INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
&P.PRTP DS C PRINTER TYPE
&P.LINES EQU X'20' LINES PRINTER TYPE
&P.PAGES EQU X'21' PAGE PRINTER TYPE 753
&P.LOADL EQU X'22' LOADABLE LINE PRINTER TYPE 753
&P.RBP EQU X'23' RBP PRINTER TYPE 753
&P.RSOPB EQU X'24' PUBLIC RSO PRINTER 753
&P.RSOPR EQU X'25' PRIVATE RSO PRINTER 753
&P.APAPR EQU X'26' APA PRINTER
&P.PUNCH EQU X'30' PUNCH TYPE 753
&P.CLTP DS C CALL TYPE
&P.NORM EQU X'01' NORMAL CALL 754
&P.RETRY EQU X'02' RETRY 754
&P.NOBUF EQU X'04' NO BUFFER 754
&P.SMLHD EQU X'08' SMALL HEADER 802
          DS CL2 RESERVED
&P.SCB@ DS A SCB ADDRESS
&P.SYIB@ DS A SYSTEM BUFFER
&P.LNPG DS F # OF LINES PER PAGE
&P.CHLN DS F # OF CHAR. PER LINE
*
* OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
*
&P.USOB@ DS A USER BUFFER ADDRESS
&P.USOBL DS F USER BUFFER LENGTH
&P.SKIP# DS F # SKIP RECORDS
&P.SYDF DS F SYSTEM DUPL FACT
&P.USDF DS F USER DUPL FACT
          DS CL12 RESERVED FOR EXTENSION
*
* INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
&P.PIR DS A PRINTER INFORMATION RECORD ADDR. 756
&P.DVTYP DS AL1 DEVICE TYPE 756
          DS XL1 HW ALIGNMENT
&P.FREQ# DS H SEPARATOR PAGES FREQUENCE # 865
*
          FREQ# IS INPUT / OUTPUT 865
&P.FWFR@ DS A ADDR OF 10 FREE FW FOR EXIT 091 865
          DS 2F RESERVED - FREE FOR USE 865
          AIF ('&L' EQ '').NOLAB
&P.LEN EQU *-&L ROUTINE EXIT P/L LENGTH
          MEXIT
.NOLAB ANOP
          AIF ('&NAME' NE '*').NOLAB1

```

```

&NAME   SETC   ''
.NOLAB1 ANOP
&P.LEN  EQU    *-&P.BEG          ROUTINE EXIT P/L LENGTH
        MEND

```

### Beispiel für den Exit 092

```

        MACRO          860          941109    32733509  SPOOL          U
&L      EX092 &D,&P
.*****
.*
.*      COPYRIGHT (C) SIEMENS AG 1990          *
.*      ALL RIGHTS RESERVED                    *
.*
.******
        LCLC  &NAME
&NAME   SETC   '&P'
*
*      SPOOL TRAILER PAGE  ROUTINE EXIT PARAMETER LIST
*
        AIF   ('&P' NE '').L5
&P      SETC  'I'
.L5     ANOP
        AIF   ('&P' NE '*').L6
&P      SETC  ''
.L6     ANOP
        AIF   (&L NE '').L01
&L      SETC  '&P.TR'
.L01    ANOP
&L      MFCHK MF=&D,DMACID=E92,ALIGN=F,SUPPORT=(C,D),ENTRY=NO,          *
        PREFIX=I,MACID=E92,DNAME=C
*
*      VERSION 850 : ADAPTATION V3 : STANDARD HEADER FOR RETURN INFO.
*
*      STANDARD HEADER RETURN INFORMATION
*
&P.BEG  DS    0H
        FHDR  MF=(C,&P),EQUATES=NO
*
*      EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
*
&P.EXNT EQU    0          NO TRAIL. PAGE OUTPUT
&P.EXSY EQU    1          SYSTEM TRAIL. UNALTERED
&P.EXAL EQU    2          ALTERNATIVE TRAILER PAGE
&P.EXBO EQU    3          SYSTEM AND ALTERNATIVE TRAILER
*      INPUT FIELDS FOR THE INPUT EXIT ROUTINE

```

```

*
&P.PRTYP DS      C                PRINTER TYPE
&P.LINES EQU    X'20'            LINES PRINTER TYPE
&P.PAGES EQU    X'21'            PAGE PRINTER TYPE                753
&P.LOADL EQU    X'22'            LOADABLE LINE PRINTER TYPE          753
&P.RBP EQU      X'23'            RBP PRINTER TYPE                753
&P.RSOPB EQU    X'24'            PUBLIC RSO PRINTER                753
&P.RSOPR EQU    X'25'            PRIVATE RSO PRINTER                753
&P.APAPR EQU    X'26'            APA PRINTERS
&P.PUNCH EQU    X'30'            PUNCH TYPE                753
&P.CLTP DS      C                CALL TYPE
&P.NORM EQU     X'01'            NORMAL CALL                754
&P.RETRY EQU    X'02'            RETRY                754
&P.NOBUF EQU    X'04'            NO BUFFER                754
&P.          DS      CL2          RESERVED
&P.SCB@ DS      A                SCB ADDRESS
&P.SYIB@ DS      A                SYSTEM BUFFER
&P.LNPG DS      F                # OF LINES PER PAGE
&P.CHLN DS      F                # OF CHAR. PER LINE
*

```

```

*   OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
*

```

```

&P.USOB@ DS      A                USER BUFFER ADDRESS
&P.USOBL DS      F                USER BUFFER LENGTH
&P.SKIP# DS      F                # SKIP RECORDS
&P.SYDF DS      F                SYSTEM DUPL FACT
&P.USDF DS      F                USER DUPL FACT
&P.          DS      CL12          RESERVED FOR EXTENSION
*

```

```

*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*

```

```

&P.PIR DS      A                PRINTER INFORMATION RECORD ADDR. 756
&P.DVTYP DS     AL1             DEVICE TYPE                756
&P.          DS      CL11          RESERVED FOR EXTENTION          756
&P.          AIF ('&L' EQ '').NOLAB
&P.LEN EQU     *-&L             ROUTINE EXIT P/L LENGTH
&P.          MEXIT
&P.          .NOLAB ANOP
&P.          AIF ('&NAME' NE '*').NOLAB1
&P.          .NOLAB1 ANOP
&P.          .NOLAB1 ANOP
&P.LEN EQU     *-&P.BEG         ROUTINE EXIT P/L LENGTH
&P.          MEND

```

## 7.4 Exit für Kunden-Filter (093)

Der Exit 093 ist ein Exit für den Dprint-Betrieb (siehe Handbuch „[Distributed Print Services \(BS2000/OSD\)](#)“).

Um Kundenwünschen und verschiedenen Dokumentformaten gerecht zu werden, die von speziellen Anwendungen oder Druckerservern generiert wurden, können mit Hilfe des Subsystems für den Exit 093 Kunden-Filter definiert werden. Diese zusätzlichen Kunden-Filter werden so genannt, da ihre Konstruktionen auf die Kundenbedürfnisse zugeschnitten sind.

Da Kunden-Filter eine höhere Priorität haben, als Standard-Systemfilter, wird bei jedem Aufruf des Filter-Mechanismus (d.h. für die beiden verschiedenen Filter-Funktionen) der Exit 093 von Dprint aufgerufen, um evtl. „Kunden“-Dokumentfilterungen durchzuführen. Eine Standard-System-Dokumentfilterung wird nur durchgeführt, wenn

- das Subsystem des Exit 093 nicht geladen ist.
- der Exit 093 über seinen Returncode diese Dokumentfilterung verlangt. Das heißt, der Exit 093 hat keine Filterung durchgeführt und erwartet die Durchführung von Dprint.

Nachfolgend werden die Voraussetzungen, die unterstützten Schnittstellen und die Einschränkungen für Entwurf und Einsatz des Exits 093 beschrieben.

### Funktionalität des Exits 093

Bei der Druck-Interoperabilität zwischen BS2000 und SINIX/UNIX-Systemen wird diese Exit-Routine jedes Mal aufgerufen, wenn eine Dokumentfilterung nötig ist. Dabei werden die folgenden Aktionen durchgeführt.

- Bei der Überprüfung der Druckanforderung:
  - Ausführen der ersten Filterfunktion, die aus der Überprüfung der Parameter des Druckauftrags besteht.
  - Ändern der Parameter für die Annahme der Druckanforderung.
  - Das in der Druckanforderung angegebene Originalformat des Dokuments in ein Dokumentformat ändern, das vom Zieldrucker unterstützt wird. Z.B. wird eine BS2000-Druckanforderung mit `DOCUMENT-FORMAT=*TEXT(...)` an eine SINIX-SPOOL-Domäne vom Standard-Systemfilter in `DOCUMENT-FORMAT=*SPECIAL-FORMAT(FORMAT-NAME=SIMPLE)` geändert. Dann kann der Wert `SIMPLE` auf das Attribut `'-ct'` der SINIX-SPOOL-Druckanforderung `'xpadding -job'` abgebildet werden.

Für diese Aktionen enthält die \$DPFL-Parameterliste, die an den Exit 093 weitergegeben wird, folgende Angaben:

- Filter-Aktionscode (\*MODIFY\_PRINT\_REQUEST)
  - Druckauftrags-Partner (\*TO\_GIP bei BS2000→ UNIX-System und \*FROM\_GIP bei UNIX-System→ BS2000)
  - Adresse der Abbildung des Druckauftrags (Beschreibung der angegebenen Operandenwerte der Druckanforderung)
  - CCSNAME-Feld, das vom Filter bei der Druck-Interoperabilität BS2000→ UNIX-System gefüllt wird und das anzeigt, ob das umgesetzte Dokument (beim zweiten Filteraufruf) im Code EBCDIC-DF04 oder ISO-8859-1 geschickt wird.
  - Filter-Kennzeichen-Feld, in den ein Filter sein Kennzeichen eintragen kann, um es beim zweiten Aufruf zu überprüfen.
- Wenn der Druckauftrag akzeptiert ist, und bevor das Dokument an den Zielsystem transferiert wird:
    - Konvertierung des Originaldokuments in ein neues, abhängig von einigen Druckanforderungs-Parametern der Abbildung des Original-Druckauftrags (d.h. Abbildung, die an den Filter für die Ausführung des ersten Aktionscodes gegeben wurde).

Für diese Aktionen enthält die \$DPFL-Parameterliste, die an den Exit 093 weitergegeben wird, folgende Angaben:

- Filter-Aktionscode (\*CONVERT-DOCUMENT)
- Druckauftrags-Partner (\*TO\_GIP bei BS2000→ UNIX-System und \*FROM\_GIP bei UNIX-System→ BS2000)
- CCSNAME-Feld, das vom Filter bei der Druck-Interoperabilität BS2000→ UNIX-System gefüllt wird und das anzeigt, ob das umgesetzte Dokument (beim zweiten Filteraufruf) im Code EBCDIC-DF04 oder ISO-8859-1 geschickt wird.
- Filter-Kennzeichen-Feld, in den ein Filter sein Kennzeichen eintragen kann, um es beim zweiten Aufruf zu überprüfen.
- Adresse der Abbildung des Original-Druckauftrags (Beschreibung der angegebenen Operandenwerte der Druckanforderung, bevor der Filter für die erste Aktion aufgerufen wird)
- Name der umgesetzten Datei, die vom Filter erstellt werden muss, und die an den Ziel-Server als Rückgabe des Filters geschickt wird.

Deshalb muss jeder Filter so implementiert werden, dass er die beiden oben beschriebenen Funktionen ausführen kann.

## Verwendung der Register

Folgenden Informationen werden an die Exit-Routine 093 übergeben:

- R0 Exit-Nummer ‚093‘
- R1 Adresse der \$DPFL-Parameterliste (siehe unten)
- R12 Adresse des TPR-Programm-Verwalters
- R13 Adresse eines Sicherungsbereichs (18 Wörter)
- R14 Adresse für indirekten Rücksprung
- R15 Adresse der Exit-Routine

## Returncode

Der Returncode des Exits 093 für die aufrufende Systemkomponente wird im Standard-Header der \$DPFL-Parameterliste abgelegt.

(SC2)	SC1	Maincode	Bedeutung
00 01	00	0000	Erfolgreiche Ausführung Erfolgreiche Ausführung Standard-Systemfilter verwenden
00 01 02 03	01	0001	Falscher Parameter Unbekannter Aktionscode Ungültige Adresse der Druckanforderung Ungültiger Ausgabedatei-Name Ungültiges Protokoll
00 01 02 03 04 05 FF	20	0002	Interner Fehler Filter-Fehler \$GETMEM-Fehler \$RETMEM-Fehler DMS-Fehler ILAM-Fehler XHCS-Fehler Anderer Fehler
00	40	0003	Filter weist Druckanforderung zurück
00	81	0004	Ungenügender Plattenspeicherplatz für Ausgabedatei



### **\$DPFL-Parameterliste**

Die \$DPFL-Schnittstelle wird verwendet, um die benötigten Informationen (z.B. Name der Ausgabedatei, Abbildung der Druckanforderung,...) an den aufgerufenen Filter zu geben und um dem Filter zu erlauben, Rückkehrinformationen an Dprint zu schicken (z.B. geänderte Abbildung der Druckanforderung, Returncode,...).

Die \$DPFL-Schnittstelle steht in Assembler und C zur Verfügung.

Schnittstellen-Name (ASS): \$DPFL

Schnittstellen-Name (C): \$DPFL.h

Schnittstellen-Titel: Aufruf-Schnittstelle der DPRINTCL-Filter

Schnittstellen-Bereich: DPRINTCL

Kompilations-Bereich: Eingeschränkt

Schnittstellen-Typ: CALL

Funktionszustand: TPR

Schnittstellen-Zweck: Schnittstelle der Interoperabilitäts-Filter

Schnittstellen-Bemerkungen: Nur Eingabe-/Ausgabe-Parameterliste (nicht MF=E)

Sprache: ASS, C

Präfix: D

Macid: PFL

## Stellung der Operanden

Offset	Identifikation	Datentyp oder Wert	Bedeutung
<b>FW</b>	<b>\$DPFL_mdl</b>	<b>STRUCT:136</b>	<b>Layout der Parameterliste</b>
000	hdr	ESMFHDR:8	Standard-Header
008	eye_catcher	CHAR:8	Eye catcher '\$DPFL004'
010	action_code MODIFY_PRINT_REQUEST  CONVERT_DOCUMENT	ENUM:1 0  255	Aktionscode Ändern der Druckanforderungs- Optionen Dokument konvertieren
011	protocol FROM_GIP TO_GIP	ENUM:1 0 128	Protokoll Vom GIP-Client Zum GIP-Server
012	reserved1	RES:2	Leerfeld
014	print_req	PTR:4	Druckanforderungs-Adresse
018	output_file	CHAR:54	Name der Ausgabedatei
04E	ccs_name	CHAR:8	Coded Character Set
056	Kennzeichen	CHAR:8	Kennzeichen des Filters
05E	reserved2	RES:42	Leerfeld

## Daten-Deklarationen für Assembler

```

*   Subcode 2 für SUCCESSFUL
DPFLSUCCESSFUL_FILTERING EQU 0           Erfolgreiches Filtern
DPFLUSE_STANDARD_FILTER EQU 1           Verwende Standardfilter
*

*   Subcode 2 für PARAMETER_ERROR
DPFLUNKNOWN_ACTION_CODE EQU 0           Unbekannter Aktionscode
DPFLNULL_REQUEST_ADDRESS EQU 1         Ungültige Adresse der Druckanf.
DPFLINVALID_OUTPUT_FILE EQU 2         Ungültiger Ausgabedatei-Name
DPFLINVALID_PROTOCOL EQU 3           Ungültiges Protokoll
*

*   Subcode 2 für INTERNAL_ERROR
DPFLFILTER EQU 0                       Filter-Fehler
DPFLGETMEM EQU 1                       $GETMEM-Fehler
DPFLRETMEM EQU 2                       $RETMEM-Fehler
DPFLDMS EQU 3                          DMS-Fehler
DPFLILAM EQU 4                          ILAM-Fehler
DPFLXHCS EQU 5                          XHCS-Fehler
DPFLOTHER EQU 255                       Anderer Fehler
*

```

```

*   Layout der Parameterliste
DPFL_MDL                DS      0F
DPFLHDR  FHDR  MF=(C,DPFL),EQUATES=NO      Standard-Header
*   Main-Returncodes
DPFLSUCCESSFUL         EQU     0           Erfolgreiches Filtern
DPFLPARAMETER_ERROR    EQU     1           Falscher Parameter
DPFLINTERNAL_ERROR     EQU     2           Interner Fehler
DPFLREQUEST_REJECTED   EQU     3           Filter weist Druckenford. ab
DPFLNO_DISK_SPACE      EQU     4           Ungenügender Plattenspeicher-
*                                           platz für Ausgabedatei
*
DPFLEYE_CATCHER        DS      CL8        Eye catcher '$DPFL004'
DPFLACTION_CODE        DS      FL1        Aktionscode
*   Unterstützte Aktionscodes
DPFLMODIFY_PRINT_REQUEST EQU     0           Druckauftrags-Optionen ändern
DPFLCONVERT_DOCUMENT   EQU    255        Dokument konvertieren
*
DPFLPROTOCOL           DS      FL1        Protokoll
*   Bekannte Protokolle
DPFLFROM_GIP           EQU     0           Von GIP-Client
DPFLTO_GIP EQU    128           An GIP-Server
*
DPFLRESERVED1          DS      XL2        Leerfeld
DPFLPRINT_REQ          DS      A          Adresse der Druckenforderung
DPFLOUTPUT_FILE        DS      CL54       Name der Ausgabedatei
DPFLCCS_NAME           DS      CL8        Coded Character Set
DPFLSIGNATURE          DS      CL8        Kennzeichen des Filters
DPFLRESERVED2          DS      XL42       Leerfeld
DPFL#                  EQU     *-DPFLHDR

```

**Daten-Deklarationen für C**

```

/* Main-Returncodes */
/* _mc_s */
#define sDPFLSUCCESSFUL 0 /* Erfolgreiches Filtern */
#define sDPFLPARAMETER_ERROR 1 /* Falscher Parameter */
#define sDPFLINTERNAL_ERROR 2 /* Interner Fehler */
#define sDPFLREQUEST_REJECTED 3 /* Filter weist Druckanf. ab */
#define sDPFLNO_DISK_SPACE 4 /* Ungenügender Plattenspei-
/* cherplatz für Ausgabedatei */

/* Subcode 2 für SUCCESSFUL */
/* ENUM _successful_sc2_s */
#define sDPFLSUCCESSFUL_FILTERING 0 /* Erfolgreiches Filtern */
#define sDPFLUSE_STANDARD_FILTER 1 /* Verwende Standardfilter */

/* Subcode 2 für PARAMETER_ERROR */
/* ENUM _parameter_sc2_s */
#define sDPFLUNKNOWN_ACTION_CODE 0 /* Unbekannter Aktionscode */
#define sDPFLNULL_REQUEST_ADDRESS 1 /* Ungült. Druckanf.-Adresse */
#define sDPFLINVALID_OUTPUT_FILE 2 /* Ungült. Ausgabedatei-Name */
#define sDPFLINVALID_PROTOCOL 3 /* Ungültiges Protokoll */

/* Subcode 2 für INTERNAL_ERROR */
/* ENUM _internal_sc2_s */
#define sDPFLFILTER 0 /* Filter-Fehler */
#define sDPFLGETMEM 1 /* $GETMEM-Fehler */
#define sDPFLRETMEM 2 /* $RETMEM-Fehler */
#define sDPFLDMS 3 /* DMS-Fehler */
#define sDPFLILAM 4 /* ILAM-Fehler */
#define sDPFLXHCS 5 /* XHCS-Fehler */
#define sDPFLOTHER 255 /* Anderer Fehler */

/* Unterstützte Aktionscodes */
/* ENUM _action_s */
#define sDPFLMODIFY_PRINT_REQUEST 0 /* Druckauft.-Optionen ändern */
#define sDPFLCONVERT_DOCUMENT 255 /* Dokument konvertieren */

/* Bekannte Protokolle */
/* ENUM _protocol_s */
#define sDPFLFROM_GIP 0 /* Von GIP-Client */
#define sDPFLTO_GIP 128 /* An GIP-Server */

```

```

/* Layout der Parameterliste */
struct sDPFL_md1 {

    /*Standard header */
    struct EMSFHDR hdr;
    char eye_catcher[8]; /* Eye catcher '$DPFL004' */
    unsigned char action_code; /* Aktionscode */
    unsigned char protocol; /* Protokoll */
    char reserved1[2]; /* Leerfeld */
    void* print_req; /* Adresse d. Druckanf. */
    char output_file[54]; /* Name der Ausgabedatei */
    char ccs_name[8]; /* Coded Character Set */
    char signature[8]; /* Kennzeichen des Filters */
    char reserved2[42]; /* Leerfeld */
};

```

### Verwaltung des Exits 093 über DSSM

Damit evtl. vorhandene Kundenfilter aktiviert werden können, muss das Subsystem geladen sein, das dem Exit 093 entspricht. Bevor dieses Subsystem geladen wird, muss es von der Systembetreuung mit SSCM im statischen Subsystemkatalog definiert werden. Der Name des Subsystems für den Ablauf des Exits 093 ist frei wählbar (z.B. DPRINTFL). Das Subsystem kann mit Hilfe der Kommandos START-SUBSYSTEM, HOLD-SUBSYSTEM, RESUME-SUBSYSTEM und STOP-SUBSYSTEM dynamisch verwaltet werden.

Weitere Informationen zur Subsystem-Definition befinden sich im Handbuch „[System Exits \(nur in Englisch\)](#)“.

### Programmierbeispiel für den Exit 093

Die Exit-Routine 093 wird zur Unterstützung von Kundenfiltern verwendet. Damit mehr als ein Kundenfilter in dem Exit 093 definiert werden kann, ist es ratsam, den Exit 093 modular aufzubauen, z.B. bestehend aus einer „Filter-Abfertigungs“-Basisroutine und aus mehreren Kundenfilter-Subroutinen.

Die „Filter-Abfertigungs“-Basisroutine kann z.B. verwendet werden, um zu überprüfen, welche Kundenfilter-Subroutine aufgerufen werden soll. Dies ist abhängig von den Dokumentformat-Attributwerten der Druckanforderung und einer internen Tabelle, die die entsprechenden „entry names“ der aufzurufenden Subroutinen enthält.

Jede Kundenfilter-Subroutine sollte so gestaltet werden, dass sie die doppelte Funktionalität der Filter entsprechend der Angaben der \$DPFL-Parameterliste erfüllt (die entsprechende Funktion wird in der \$DPFL-Parameterliste bei jedem Aufruf des Exits 093 veranlasst):

- Ändern des Druckauftrags und
- Umsetzen des Dokuments.

Nachfolgendes Beispiel gibt einen Überblick über die Implementierung des Exits 093 in C. Die Verwendung der Sprache C erfordert die Implementierung eines kleinen Assembler-nach-C-Moduls (im Beispiel EXIT093 genannt), das von der Exit-Schnittstelle \$SYSX aufgerufen wird. Der Benutzer braucht nur das nachfolgend gezeigte Gerüst verwenden und nach Belieben um die gewünschten C-Funktionen ergänzen. Die HEADER-Datei <msg7x.h> und die Rücksende-Routine müssen für die Ausgabe von Fehlermeldungen vom Benutzer erstellt werden.

#### *EXIT093-Schnittstellenmodul Assembler-nach-C*

```

                PRINT NOGEN
EXIT093 @ENR  TYP=E,ENV=SPLSPEC,RETURNS=NO,TITLE=NO,LOCAL=WADF
EXIT093 AMODE ANY
EXIT093 RMODE ANY
                ST   R1,INPL@           $DPFL-Adresse sichern
                LA   R1,INPL@           Adressen d. gesicherten $DPFL-Adresse holen
                L    R15,=V(EXIT93C)
                BASR R14,R15            C-Programm EXIT93C aufrufen
                @EXIT ,                 Zum Aufrufer zurückkehren
                @END ,
WADF @PAR  D=YES
INPL@ DS   A
WADF @PAR  LEND=YES
                END

```

#### *EXIT93C-C-Funktion (durch EXIT093-Modul aufgerufen)*

```

/*-----*/
/* #include AND LOCAL DEFINITIONS */
/*-----*/
#include <stdio.h>
#include <string.h>
#include <esmfhdr.h>
#include <prntdpc.h>
#include <prntdoc.h>
#include <$dpfl.h>
#include <msg7x.h>
.
.
.
struct p1_md1 { struct sDPFL_md1 *dpfl;
.

```

```

.
.
};

/*-----*/
/*  INTERNAL CALL PROTOTYPES          */
/*-----*/
static void to_gip_modify (struct pl_md1 *);
static void to_gip_convert (struct pl_md1 *);
static void dispatcher (struct pl_md1 *);
static void send_reject (char, int);
.
.
.
/*-----*/
/*  STATIC DATA PART                */
/*-----*/
.
.
.
/*-----*/
/*  CODE PART                        */
/*-----*/
.
.
.
/*-----*/
/*  to_gip_modify                    */
/* -----                          */
/*                                  */
/* Diese Routine führt den ersten Aufruf des Fil-      */
/* ters für die Gültigkeitsprüfung und die Änderung  */
/* von BS2000-nach-UNIX-System-Druckanforderungen durch. */
/*-----*/

void to_gip_modify (struct pl_md1 *pl)
{
    struct sDPFL_md1 *dpfl = pl->dpfl;
    struct PRNTDOC_pl_md1 *prntdoc = dpfl->print_req;
    struct PRNTDPC_pl_md1 *prntdpc = prntdoc->dppar;
.
.
/* -----
   Abhängig von den Druckauftrags-Parametern
   werden andere Funktionen (intern oder ex-
   tern) aufgerufen, um die entsprechende
   Prüfung und Änderung durchzuführen.
   ----- */

```

```

.
.
}

/*-----*/
/* to_gip_convert */
/* ----- */
/* */
/* Diese Routine führt den zweiten Aufruf des Fil- */
/* ters für die Bearbeitung der Datei und Erstell- */
/* lung einer Ausgabedatei für BS2000-nach- */
/* UNIX-System-Druckanforderungen durch */
/*-----*/

void to_gip_convert (struct pl_md1 *pl)
{
    struct sDPFL_md1 *dpfl = pl->dpfl;
    struct PRNTDOC_pl_md1 *prntdoc = dpfl->print_req;
    struct PRNTDPC_pl_md1 *prntdpc = prntdoc->dppar;
.
.
    /* -----
       Abhängig von den Druckauftrags-Parametern
       werden andere Funktionen (intern oder ex-
       tern) aufgerufen, um die entsprechende
       Ausgabedatei zu erstellen.
       ----- */
.
.
}

/*-----*/
/* Dispatching routine */
/* ----- */
/* */
/* Abschicken der Druckanforderung abhängig von */
/* Protokoll und Aktionscode sowie Prüfung nach */
/* Parameterfehlern. */
/*-----*/

void dispatcher (struct pl_md1 *pl)
{
    struct sDPFL_md1 *dpfl = pl->dpfl;
    if (dpfl->print_req == NULL) {
        dpfl->hdr.maincode = sDPFLPARAMETER_ERROR;
        dpfl->hdr.subcode2 = sDPFLNULL_REQUEST_ADDRESS;
        send_reject (MSG7X_DEST_SYSOUT, NULL_REQUEST_ADDRESS);
        send_reject (MSG7X_DEST_SYSOUT, NULL_REQUEST_ADDRESS);
        return;
    }
}

```



```

}
switch (dpfl->protocol) {
    case    sDPFLFROM_GIP :
        /* -----
        Übermittler der UNIX-System-nach-BS2000-Druck-
        anforderungen: nachfolgendes Beispiel
        zeigt einen Exit 093. der nichts anderes
        macht, als lediglich über seinen Return-
        code den mit DPRINTCL ausgelieferten Sy-
        stem-Standardfilter aufzurufen.
        ----- */

        dpfl->hdr.maincode = sDPFLSUCCESSFUL;
        dpfl->hdr.subcode2 =
            sDPFLUSE_STANDARD_FILTER;
        break;

    case    sDPFLTO_GIP :
        /* -----
        Übermittler der BS2000-nach-UNIX-System-Druck-
        anforderungen: im nachfolgenden Beispiel
        wird der Actionscode überprüft und die
        entsprechenden Funktionen aufgerufen.
        ----- */

        switch (dpfl->action_code) {
            case    sDPFLMODIFY_PRINT_REQUEST :
                to_gip_modify (p1);
                break;

            case    sDPFLCONVERT_DOCUMENT :
                to_gip_convert (p1);
                break;

            default:
                dpfl->hdr.maincode = sDPFLPARAMETER_ERROR;
                dpfl->hdr.subcode2 =
                    sDPFLUNKNOWN_ACTION_CODE;
                send_reject (MSG7X_DEST_SYSOUT, INVALID_ACTION_CODE);
                break;
        }
        break;

    default :
        dpfl->hdr.maincode = sDPFLPARAMETER_ERROR;
        dpfl->hdr.subcode2 = sDPFLINVALID_PROTOCOL;
        send_reject (MSG7X_DEST_SYSOUT, INVALID_PROTOCOL);
        break;
}

```

```

    }
}
/*-----*/
/* EXIT 93 C entry point */
/* ----- */
/* */
/* Dies ist der Entry-Point des Exit 093 in das C- */
/* Modul. Es wird vom Schnittstellen-Programm im */
/* EXIT093 aufgerufen, die den $SYSX-Aufruf in ei- */
/* nen Standard-Aufruf umwandelt, der in C sein muss */
/*-----*/

extern void EXIT93C (struct sDPFL_md1 *dpfl)
{
    struct pl_md1 pl;

    dpfl->hdr.maincode = sDPFLSUCCESSFUL;
    dpfl->hdr.subcode2 = sDPFLSUCCESSFUL_FILTERING;

    pl.dpfl = dpfl;
    dispatcher (&pl);
}

```

### Sicherheitsaspekte

Exit-Routinen laufen wie die aufrufenden Systemkomponenten im privilegierten Zustand ab. Bei ihrer Programmierung ist deshalb besondere Sorgfalt vonnöten. Insbesondere können die Verwendung von TU-Makros oder andere Programmierfehler einen Task-Abbruch oder gar einen Systemausfall zur Folge haben. Fehler in Exit-Routinen werden wie Systemfehler behandelt. Einzelheiten über die unbedingt zu beachtenden Regeln und Einschränkungen befinden sich im Handbuch „[System Exits \(nur in Englisch\)](#)“.

## 7.5 Ressourcen-Routine (094)

Dieser SPOOL-Exit ermöglicht es der Systembetreuung, eine Anwenderoutine zu erstellen, die zwei Funktionen innerhalb einer Client-Server-Umgebung bei Einsatz des Subsystems Distributed Print Services erfüllen soll:

- die Konvertierung von Client-Ressourcen auf Server-Ressourcen

Diese Funktion modifiziert die Werte einiger Felder des SCB (z.B. FORM, DIA).

Die Eingabe besteht aus dem Standardheader, der Adresse des SCB und dem Funktionscode F0.

Die Ausgabe besteht aus dem Returncode im Standardheader und dem modifizierten SCB.

- die Konvertierung von Server-Ressourcen auf Client-Ressourcen

Diese Funktion stellt die Originalwerte der Felder des SCB (z.B. FORM, DIA) wieder her.

Die Eingabe besteht aus dem Standardheader, der Adresse des SCB und dem Funktionscodes F1.

Die Ausgabe besteht aus dem Returncode im Standardheader und dem modifizierten SCB.

Der Exit 094 muss auf dem Client- und dem Server-Rechner gestartet werden. Anderenfalls wird der SPOOLOUT-Auftrag mit der Meldung SCP089 abgewiesen. Der Exit wird während der Validierung des Druckauftrags aufgerufen.

Der Einsatz der Exit-Routine mit dem Funktionscode F1 zur Wiederherstellung des Originalnamens im SCB ist optional. Lediglich der Returncode muss gesetzt werden. Die vom Server durchgeführte Übersetzung ist jedoch auch in diesem Fall transparent für den Anwender am Client-Rechner. Bei allen SHOW-PRINT-JOB-Kommandos ist die Ausgabe entsprechend modifiziert. In jedem Fall werden dem Cluster-Verwalter die übersetzten Werte angezeigt.

Als Voreinstellung wird das Schlussblatt mit den Werten des Servers ausgedruckt. Will der Anwender sein Schlussblatt mit den Werten des Clients ausdrucken, muss er einen entsprechenden Trailer-Exit mit Aufruf des Exits 094 erstellen.

Tritt während der Verarbeitung des Funktionsaufrufs F0 ein Fehler auf (Returncode ungleich 0 im Standardheader), wird der SPOOLOUT-Auftrag mit der Fehlermeldung SCP1090 abgewiesen.

Tritt während der Verarbeitung des Funktionsaufrufs F1 ein Fehler auf (Returncode ungleich 0 im Standardheader), wird eine Fehlermeldung an der Bedienstation ausgegeben, der SPOOLOUT-Auftrag jedoch fortgesetzt.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX094-Parameterliste)

Die Returncodes werden im Standardheader eingetragen.

Standard-  
header:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX094 wird im Standardheader ein Returncode übergeben.

aa=Maincode1; bb=Maincode2; cc=Subcode1; dd=Subcode2

X'aa'	Erläuterung
00	Exit fehlerfrei ausgeführt
01	Exit nicht fehlerfrei ausgeführt

## DSECT

Eine DSECT/CSECT für die Parameterliste kann mit dem Makro EX094 erzeugt werden, wobei "prefix" max. 3 Zeichen lang sein darf.

name EX094 D/C ,[prefix]

```

      SPOOLEX4 EX094 D
1 SPOOLEX4 DSECT ,           RENAMING OF CLIENT RESOURCE EXIT P/L
1 E94EXBEG DS      0H
1          FHDR MF=(C,E94),EQUATES=NO
2          DS      0A
2 E94FHE   DS      0XL8      0   GENERAL PARAMETER AREA HEADER
2 *
2 E94IFID  DS      0A        0   INTERFACE IDENTIFIER
2 E94FCTU  DS      AL2      0   FUNCTION UNIT NUMBER
2 *
2 *
2 *
2 *
2 *
2 E94FCT   DS      AL1      2   FUNCTION NUMBER
2 E94FCTV  DS      AL1      3   FUNCTION INTERFACE VERSION NUMBER
2 *
2 E94RET   DS      0A        4   GENERAL RETURN CODE
2 E94SRET  DS      0AL2     4   SUB RETURN CODE
2 E94SR2   DS      AL1      4   SUB RETURN CODE 2
2 E94SR1   DS      AL1      5   SUB RETURN CODE 1
2 E94MRET  DS      0AL2     6   MAIN RETURN CODE
2 E94MR2   DS      AL1      6   MAIN RETURN CODE 2

```

```

2 E94MR1 DS AL1 7 MAIN RETURN CODE 1
2 E94FHL EQU 8 8 GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *
1 * EQUATES FOR EXIT RETURN INFORMATION IN &P.RET
1 *
1 E94EXNOK EQU 1 EXIT NOK
1 E94EXOK EQU 0 EXIT OK
1 E94INFOR DC X'00' INFORMATION GIVEN TO EXIT
1 E94CLTSV EQU C'0' TRANS. CLIENT RESSOURCE TO SERV
1 E94SVTCL EQU C'1' TRANS. SERVER RESSOURCE TO CLI.
1 E94UNUS DC XL3'00' **** UNUSED ****
1 E94SCBAD DC A(0) ADDRESS OF SCB
1 DS CL20 RESERVED FOR EXTENTION
1 E94PLEN EQU *-E94EXBEG LENGTH OF THE PARAMETER LIST

```

### Beispiel für den Exit 094

```

MACRO 050 960829 SPOOL U
&NAME EX094 &D,&P
.*****
.* BEGIN-INTERFACE EX094
.*
.* TITLE (/ ex094 /)
.* NAME EX094
.* DOMAIN SPOOL
.* LANGUAGE ASS
.* COPYRIGHT (C) Siemens AG 1994
.* ALL RIGHTS RESERVED
.* COMPILATION-SCOPE USER
.* INTERFACE-TYPE LAYOUT
.* RUN-CONTEXT TPR
.*
.* PURPOSE (/ translation of resource (client or server) /)
.*
.* SYNTAX (/ Syntax Variant 1:
.* &NAME EX094 &D,&P
.* /)
.*
.* REMARKS (/ corresponding include : SSVEX094 /)
.*****
.*
.* END-INTERFACE EX094.
.*****
AIF ('&P' NE '').AO
&P SETC 'E94'
.AO ANOP

```

```

        AIF ('&NAME' NE '').A1
&NAME  SETC 'CLREXPL'
.A1    ANOP
        AIF ('&D' EQ 'D').A2
&NAME  DS   OF          RENAMING OF CLIENT RESOURCE EXIT P/L
        AGO  .A3
.A2    ANOP
&NAME  DSECT ,          RENAMING OF CLIENT RESOURCE EXIT P/L
.A3    ANOP
&P.EXBEG DS   OH
        FHDR MF=(C,&P),EQUATES=NO
*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.RET
*
&P.EXNOK EQU   1          EXIT NOK
&P.EXOK  EQU   0          EXIT OK
&P.INFOR DC   X'00'      INFORMATION GIVEN TO EXIT
&P.CLTSV EQU  C'0'       TRANS. CLIENT RESSOURCE TO SERV
&P.SVTCL EQU  C'1'       TRANS. SERVER RESSOURCE TO CLI.
&P.CALCL EQU  C'2'       TRANS. CALL ON CLIENT
&P.UNUS  DC   XL3'00'    **** UNUSED ****
&P.SCBAD DC   A(0)      ADDRESS OF SCB
        DS   CL20        RESERVED FOR EXTENTION
&P.PLEN  EQU  *-&P.EXBEG LENGTH OF THE PARAMETER LIST
        MEND

```

## 7.6 Ausgabe-Routine (095)

Dieser SPOOL-Exit ermöglicht es der Systembetreuung, eine Anwenderoutine zu erstellen, die bei jeder Ausgabeoperation des SPOOL aufgerufen wird. Nur bei der Fehlerbehandlung erfolgt kein Aufruf. Die Exit-Routine wird aufgerufen, nachdem ein Kanalprogramm erstellt wurde und bevor dieses zum Drucker gesendet wird. Die Exit-Routine kann für Zeilen- und Laserdrucker verwendet werden. Die APA-Drucker, RSO, Remote-Batch-Verarbeitung, Magnetbänder werden nicht unterstützt.

Für die Exit-Routine bestehen folgende Möglichkeiten:

- das Kanalprogramm zu akzeptieren
- das Kanalprogramm zu verändern
- das Kanalprogramm zu unterdrücken.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX095-Parameterliste)

Die Returncodes werden im Standardheader eingetragen.

Standard-  
header:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX095 wird im Standardheader ein Returncode übergeben.

aa=Maincode1; bb=Maincode2; cc=Subcode1;  
dd=Subcode2

X'aa'	Erläuterung
00	Kanalprogramm ausgegeben
04	Kanalprogramm nicht ausgegeben
0C	Kanalprogramm nicht ausgeben und EOF

### *Hinweise*

- Die Exit-Routine hat Zugriff zu den SPOOL-eigenen Datenstrukturen. Diese Datenstrukturen sind versionsabhängig.
- Wird das Kanalprogramm verändert oder nicht ausgegeben, erfolgt keine Berichtigung des SPOOL-Abrechnungssatzes.
- Für die von der Exit-Routine übergebenen Daten wird keine "Code-Translation" durchgeführt.
- Wird das Kanalprogramm verändert, sind folgende Regeln zu beachten:
  - Das Kanalprogramm und der Datenpuffer dürfen nicht erweitert werden.
  - Werden die Daten, die von einem Kanalbefehl adressiert sind, verkürzt, muss auch die Datenlängenangabe des Kanalbefehls korrigiert werden.

- Wird das Kanalprogramm verkürzt, muss auch der "Kettenende-Indikator" korrigiert werden.
- Die Datenadresse kann verändert werden, muss aber immer im SPOOL-Ausgabebereich liegen. Die Adresse "Beginn Ausgabebereich" ist im ersten Kanalbefehl der Kette enthalten. Die Adresse "Ende Ausgabebereich" ergibt sich aus der Datenlänge plus Datenadresse des letzten Kanalbefehls.

## DSECT

Eine DSECT/CSECT für die Parameterliste kann mit dem Makro EX095 erzeugt werden, wobei "prefix" max. 3 Zeichen lang sein darf.

```
name EX095 D/C ,[prefix]
```

```

    SPOOLEX5 EX095 D
1 SPOOLEX5 DSECT ,           I/O   EXIT ROUTINE P/L
1 SPOEXBEG DS    0H
1          FHDR MF=(C,SPO),EQUATES=NO
2          DS    0A
2 SPOFHE   DS    0XL8        0   GENERAL PARAMETER AREA HEADER
2 *
2 SPOIFID  DS    0A          0   INTERFACE IDENTIFIER
2 SPOFCTU  DS    AL2         0   FUNCTION UNIT NUMBER
2 *
2 *                               BIT 15   HEADER FLAG BIT,
2 *                               MUST BE RESET UNTIL FURTHER NOTICE
2 *                               BIT 14-12 UNUSED, MUST BE RESET
2 *                               BIT 11-0   REAL FUNCTION UNIT NUMBER
2 SPOFCT   DS    AL1         2   FUNCTION NUMBER
2 SPOFCTV  DS    AL1         3   FUNCTION INTERFACE VERSION NUMBER
2 *
2 SPORET   DS    0A          4   GENERAL RETURN CODE
2 SPOSRET  DS    0AL2        4   SUB RETURN CODE
2 SPOSR2   DS    AL1         4   SUB RETURN CODE 2
2 SPOSR1   DS    AL1         5   SUB RETURN CODE 1
2 SPOMRET  DS    0AL2        6   MAIN RETURN CODE
2 SPOMR2   DS    AL1         6   MAIN RETURN CODE 2
2 SPOMR1   DS    AL1         7   MAIN RETURN CODE 1
2 SPOFHL   EQU    8          8   GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *
1 *   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
1 *
1 SPOEXIO  EQU    0          SEND I/O REQUESTED
1 SPOEXNI  EQU    4          DO NOT SEND I/O
1 SPOEXEO  EQU    12         NO I/O DECLARE EOF
1 SPOPRYP  DC    AL1(0)     PRINTER TYPE
1 SPOLINES EQU    32         LINES PRINTER TYPE           751
1 SPOPAGES EQU    33         PAGE PRINTER TYPE             751
1 SPOLOADL EQU    34         LOADABLE LINE PRINTER TYPE    751

```



1 SPOINFOR DC	X'00'	INFORMATION GIVEN TO EXIT
1 SPONOSCB EQU	X'01'	IND. SPOSCBAD NOT FILLED
1 SPOUNUS DC	XL2'00'	**** UNUSED ****
1 SPOCHADD DC	A(0)	CCW'S CHAIN OR BUFFER ADDRESS
1 SPOSCBAD DC	A(0)	ADDRESS OF SCB
1 SPODVTYP DS	AL1	DEVICE TYPE
1 SPOPRTMN DS	CL2	PRINTER MNEMONIC
1 DS	CL20	RESERVED FOR EXTENTION
1 SPOPLLEN EQU	*-SPOEXBEG	LENGTH OF THE PARAMETER LIST

Das Feld SPOCHADD enthält die Adresse des Kanalprogramms. Das Format eines Kanalbefehls ist unabhängig von Kanal- und Maschinentyp.

op	fl	count	address
----	----	-------	---------

op = Operationscode  
 fl = Flagbyte  
 count = Datenlänge  
 address = virtuelle Datenadresse

#### *Hinweis*

Die Operationscodes des Kanalbefehls sind abhängig von Kanal- und Druckertyp und werden von SPOOL nach der Rückkehr aus der Exit-Routine nicht überprüft.

Das Feld &P.NOSCB (siehe Parameterliste) zeigt an, dass das Feld &P.SCBAD keine SCB-Adresse enthält und somit nicht verwendet werden kann. Das kommt vor, wenn der SPOOL Ein-/Ausgabeoperationen durchführt, die keinem Job zugeordnet sind.

Das Feld &P.DVTYP enthält den gleichen detaillierten Gerätetyp, der in der SPOOL-Parameterdatei benutzt wird. Für die Auflösung ist der Makro SPSDTYP zu verwenden.

Das Feld &P.PRTMN enthält die mnemotechnische Bezeichnung des Druckers.

## 7.7 Exit für Server-Auswahl (096)

Mit diesem Exit kann die SPOOL-Verwaltung (TSOS bzw. PRINT-SERVICE-ADM) eine Server-Auswahl durch Kombination bestimmter Auswahl-Kriterien vornehmen. Die Server-Auswahl kann dabei einer individuellen Entscheidungstabelle angepasst sein, die beispielsweise vorsieht, dass die Aufträge von Anwender A mit dem Papierformat A auf den Server X geleitet werden sollen, die Aufträge von Anwender A mit dem Papierformat B auf Server Y u.s.w.

Der Exit 96 wird nicht nur in einer Dprint-Umgebung mit einer Liste von Servern aufgerufen, sondern auch (falls nötig) für den lokalen Spool, wobei die Serverliste nur ein einziges Element enthält, dessen Name aus acht Leerzeichen besteht.

Der Exit 096 muss am Client-Rechner gestartet werden. Er wird während der Validierung des Druckauftrags aufgerufen.

Bei Änderungen der Server-Liste muss auf deren Konsistenz geachtet werden. Der vom SPOOL-Verwalter für einen Auftrag ausgewählte Server muss in der dem Exit übergebenen Server-Liste enthalten sein. Um diese Anforderung sicherzustellen, kann der SPOOL-Verwalter

- den gewünschten Server an die erste Position der Liste setzen und die anderen Server beibehalten
- den gewünschten Server an die erste Position der Liste setzen und die Anzahl der in der Liste verfügbaren Server auf 1 setzen.

Soll ein möglicherweise verteilter Druckauftrag zum lokalen Spool umgeleitet werden, muss im Aufruf der Exit-Routine für die Dprint-Umgebung SERVER# auf Null gesetzt werden. Im nun nachfolgenden Aufruf für den lokalen Spool muss die Serverliste bleiben, wie sie ist (ein Element bestehend aus acht Leerzeichen), sonst wird der Druckauftrag komplett abgewiesen.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX096-Parameterliste)

Die Returncodes werden im Standardheader eingetragen.

Standard-  
header:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX096 wird im Standard-  
header ein Returncode übergeben.

aa=Maincode1; bb=Maincode2; cc=Subcode1;  
dd=Subcode2

X'aa'	Erläuterung
00	Exit fehlerfrei ausgeführt
01	Exit nicht fehlerfrei ausgeführt

**DSECT**

Eine DSECT/CSECT für die Parameterliste kann mit dem Makro EX096 erzeugt werden, wobei "prefix" max. 3 Zeichen lang sein darf.

name EX096 D/C ,[prefix]

```

      SPOOLEX6 EX096 D
1 SPOOLEX6 DSECT ,          SERVER SELECTION EXIT P/L
1 E96EXBEG DS      0H
1          FHDR MF=(C,E96),EQUATES=NO
2          DS      0A
2 E96FHE   DS      0XL8          0  GENERAL PARAMETER AREA HEADER
2 *
2 E96IFID  DS      0A          0  INTERFACE IDENTIFIER
2 E96FCTU  DS      AL2          0  FUNCTION UNIT NUMBER
2 *
2 *
2 *
2 *
2 E96FCT   DS      AL1          2  FUNCTION NUMBER
2 E96FCTV  DS      AL1          3  FUNCTION INTERFACE VERSION NUMBER
2 *
2 E96RET   DS      0A          4  GENERAL RETURN CODE
2 E96SRET  DS      0AL2         4  SUB RETURN CODE
2 E96SR2   DS      AL1          4  SUB RETURN CODE 2
2 E96SR1   DS      AL1          5  SUB RETURN CODE 1
2 E96MRET  DS      0AL2         6  MAIN RETURN CODE
2 E96MR2   DS      AL1          6  MAIN RETURN CODE 2
2 E96MR1   DS      AL1          7  MAIN RETURN CODE 1
2 E96FHL   EQU     8            8  GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *
1 *   EQUATES FOR EXIT RETURN INFORMATION IN &P.RET
1 *
1 E96EXNOK EQU     1            EXIT NOK
1 E96EXOK  EQU     0            EXIT OK
1 E96SCBAD DC      A(0)        ADDRESS OF SCB
1 E96SVLAD DC      A(0)        ADDRESS OF SERVER LIST
1          DS      CL12        RESERVED FOR EXTENTION
1 E96PLEN  EQU     *-E96EXBEG  LENGTH OF THE PARAMETER LIST
1 E96SVL   DSECT ,          SERVER LIST DESCRIPTION
1 E96SV#   DS      F            SERVERS#
1 E96SVSZ  DS      F            SIZE
1 E96SV    DS      0XL20       SERVER(SERVERS#) (1 ENTRY)
1 E96SVN   DS      CL8         NAME
1 E96APR#  DS      FL4         PRINTERS#
1 E96DT    DS      XL8         DEVICE_TYPES

```

## 7.8 Exit zum Steuern des File Transfers für Dprint-Aufträge (097)

Der Exit 097 ist in SPOOL verankert. Er ist jedoch nur für eine Dprint-Umgebung relevant, die über das Produkt Distributed Print Services angeboten wird. Die Aufgaben des Exit 097 sind:

- Zu verhindern, dass der Server durch zu viele Druckaufträge, die im gleichzeitig im File Transfer sind, überlastet wird.
- Zu verhindern, dass Dprint die FT-Ressourcen auf der Client-Seite für sich alleine beansprucht.

Um diese Aufgaben zu erfüllen, muss der Exit 097 auf der Server-Seite aktiv sein. Der Exit reguliert die Anzahl der Druckaufträge, die gleichzeitig zum Server übertragen werden. Dazu wird für jeden Rechner im Cluster eine maximale Anzahl der Druckaufträge festgelegt, die gleichzeitig übertragen werden dürfen. Der Grenzwert für den Server ist auf diese Weise die Summe aller Grenzwerte der Clients.

Die maximale Anzahl der Druckaufträge, die gleichzeitig von einem Client-Rechner zum Server übertragen werden dürfen, ist die Summe aller Grenzwerte, die auf den Servern dieses Clusters für diesen Client-Rechner definiert wurden.

Alle diese Grenzwerte sind mit dafür vorgesehenen Kommandos SHOW-DPRINT-FT-LIMIT, MODIFY-DPRINT-FT-LIMIT, ADD-DPRINT-FT-LIMIT und REMOVE-DPRINT-FT-LIMIT sichtbar und einstellbar.

### Hintergrund

Wurde ein von einem Client abgegebener Dprint-Auftrag von einem fernen Server angenommen, befindet er sich in der SPOOL-Auftragswarteschlange des Servers im Status WFT (wait file transfer). Auf die Anforderung der Dprint-FT-Einheit wählt SPOOL einen Auftrag aus der Warteschlange aus und übergibt ihn an Dprint. Dabei wird der Auftrag gewählt, der die höchste Priorität hat und dessen Original-Host (d.h. der Client-Host, der den Auftrag abgegeben hat) verfügbar ist und freie Betriebsmittel besitzt.

Hat die Dprint-FT-Einheit den Auftrag erhalten, informiert sie den Client-Host. Dann initiiert dieser den File Transfer der zu druckenden Datei und evtl. der Datei, die die Ressourcen für den Ausdruck enthält.

Da die Dprint-FT-Einheit Aufträge ohne Unterbrechung anfordert, werden alle Aufträge, die vom Server angenommen wurden und FT benötigen, sofort an FT geleitet. Ist die Anzahl von Aufträgen sehr hoch und ist der Server nicht leistungsfähig genug, führt dies zu Performance-Problemen auf der Server-Seite und zu einer alleinigen Inanspruchnahme der File Transfer-Ressourcen auf der Client-Seite.

Der Exit 097 ist im SPOOL auf der Server-Seite lokalisiert und greift dort ein, wo ein Auftrag für die Dprint-FT-Einheit ausgewählt wird.

Das Exit-097-Subsystem muss auf dem Server-Rechner gestartet werden. Es kann selbst dann geladen werden, wenn SPOOL oder Dprint nicht gestartet sind. In diesem Zustand stellt das Subsystem nur die Schnittstellen-Kommandos bereit. Wenn Dprint auf dem Server geladen wird, startet der Regulierungsmechanismus. Er wird alle 30 Sekunden durch die Dprint FT-Einheit angestossen, die einen neuen Auftrag zur Verarbeitung anfordert.

### **Exit-Verarbeitung**

Für jeden Rechner im Cluster errechnet das Exit-Subsystem die Anzahl der Druckaufträge, die auf den File Transfer warten, und die Anzahl der Druckaufträge, die gerade übertragen werden. Dabei gibt es einen Unterschied zwischen dem Status, den das Kommando SHOW-PRINT-JOB-STATUS ausgibt, und dem Status, den das Exit 097-Subsystem intern verwendet. Ein Druckauftrag ist ab dem Augenblick im Status WFT, in dem er vom Server akzeptiert wurde, bis zu dem Augenblick, in dem der Server die Information empfängt, dass der Client den File Transfer (FT NCOPY) korrekt eingeleitet hat. In diesem Moment wechselt der Status des Druckauftrags von WFT auf FT.

Für das Exit 097-Subsystem wartet ein Druckauftrag auf den File Transfer, vom Augenblick des Akzeptierens durch den Server bis die Verarbeitung durch Dprint Server FT auf dem Server eingeleitet wurde. Ab diesem Augenblick wird der Druckauftrag für das Subsystem übertragen.

Das Ergebnis der Berechnung wird in einer internen Host-Tabelle abgelegt. Zu jedem Rechnereintrag gehört eine Zahl, die die maximale Anzahl von Druckaufträgen angibt, die gleichzeitig von diesem Rechner übertragen werden dürfen (FT-Limit).

Wenn das Subsystem während der Exit-Verarbeitung auf einen Druckauftrag stößt, der von einem Rechner stammt, der noch nicht in der Tabelle eingetragen ist, wird automatisch ein neuer Eintrag generiert. In diesem Fall wird dem Rechner ein vorgegebener Wert für FT-Limit zugewiesen. Nach dem Laden des Subsystems für den Exit 097 liegt dieser Vorgabewert bei 10.

Ein neuer Eintrag kann explizit mit den weiter unten beschriebenen Kommandos hinzugefügt werden. In diesem Fall muss der Wert für FT-Limit angegeben werden.

Die Exit-Verarbeitung verhindert die Initialisierung des File Transfers für Druckaufträge von Rechnern, die ihre maximale Anzahl an Druckaufträgen gleichzeitig übertragen.

## Anwendungshinweise

- Der [Abschnitt „Unterstützende Kommandos für den Exit 097“ auf Seite 593](#) beschreibt Kommandos, mit denen Sie die interne Host-Tabelle eines Dprint-Servers verwalten können. Diese Kommandos sind in einer Syntaxdatei definiert, die Sie zusammen mit dem Subsystem für den Exit 097 erhalten.
- Die interne Host-Tabelle existiert nur so lange, wie das Exit-Subsystem im Serverrechner geladen ist. Das bedeutet, dass die Host-Tabelle verschwindet, wenn das Exit-Subsystem gestoppt wird.  
Sobald das Exit-Subsystem neu gestartet wird, muss die Host-Tabelle neu aufgebaut werden. Entweder wird die Tabelle automatisch vom Exit wiedererstellt und alle Werte für FT-Limit werden auf den Vorgabewert zurückgesetzt und alle individuellen Werte für FT-Limit sind gelöscht, oder Sie schreiben eine Prozedur, die für jeden Rechner ein Kommando ADD-DPRINT-FT-LIMIT mit dem Wert für FT-Limit enthält. In diesem Fall ist es notwendig, dass zuerst das Exit-Subsystem geladen wird. Danach muss Ihre Prozedur die Host-Tabelle mit Daten füllen. Erst dann kann der Dprint-Server mit den individuellen Werten für FT-Limit gestartet werden.

## EXIT-Spezifikation

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = Adresse der EX097-Parameterliste

Die Parameterliste enthält folgende Felder:

E97HSTL@ = Hostlist-Adresse

Adresse einer Tabelle, die die Namen der verfügbaren und nicht gesättigten Hosts enthält

E97HST# = Anzahl der Einträge in der Host-Tabelle

E97HSTL = Länge eines Eintrags in der Host-Tabelle

E97SVT@ = Adresse der SPOOL-Vector-Tabelle

Folgende Ausgaben werden von der Exitroutine geliefert:

Returncode: Der Returncode des Exits 097 wird im Standard-Header der Parameterliste abgelegt.

Standard-  
header:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX096 wird im Standard-header ein Returncode übergeben.

aa=Maincode1; bb=Maincode2; cc=Subcode1;  
dd=Subcode2

X'aa'	Erläuterung
00	Exit fehlerfrei ausgeführt
01	Exit nicht fehlerfrei ausgeführt

E97HOK# = Anzahl in der Eingabe-Host-Liste gebliebener Hosts

E97WFT# = Anzahl der Aufträge im Zustand WFT für die Hosts, die in der Eingabe-Host-Liste geblieben sind

E97FT# = Anzahl der Aufträge im Zustand FT für die Hosts, die in der Eingabe-Host-Liste geblieben sind

Eingabe-Host-Liste = Eingabeliste, aus der einige Einträge entfernt wurden (= mit binären Nullen aufgefüllt)

**DSECT**

```

SP00LEX7 EX097 D
E97EXBEG DS    0H
          FHDR MF=(C,E97),EQUATES=NO
*
*   EQUATES FOR EXIT RETURN INFORMATION IN E97RET
*
E97EXNOK EQU    1           EXIT NOK
E97EXOK  EQU    0           EXIT OK
E97UNUS  DC     XL4'00'     ****  UNUSED  ****
*
E97HSTL@ DS     A           ACCESSIBLE HOST LIST
E97SVT@  DS     A           SVT ADDRESS
E97HST#  DS     H           NBR OF ENTRIES IN THE LIST
E97HSTL  DS     H           LENGTH OF AN ENTRY IN THE LIST
*
E97HOK#  DS     H           OUTPUT AREA = NBR OF CORRECT HST
E97WFT#  DS     H           TOTAL NBR OF JOB IN WFT
E97FT#   DS     H           TOTAL NBR OF JOB IN FT
          DS     CL20        RESERVED FOR EXTENSION
E97PLEN  EQU    *-E97EXBEG  LENGTH OF THE PARAMETER LIST
          MEND

```



## Erläuterung und Beispiel des Exit 097

Der Exit 097 erstellt eine interne Host-Tabelle und berechnet für jeden Host die Anzahl der Aufträge im Status WFT und FT. Für jeden Host wird ein Maximalwert für gleichzeitige Aufträge in File Transfer gesetzt. Dieser Maximalwert ist im Exit-Modul als konstanter Wert kodiert, sein Wert ist 10. Der Wert ist für jeden Host gleich.

Dieser Wert kann mit einem Rep in dem Exit-Modul geändert werden. Beispiel: für den Maximalwert 15 muss in EXIT#097+X'210' X'000F' gesetzt werden (alter Code = X'000A').

Der Wert kann geändert werden, während die Bearbeitung läuft.

Hat ein Host aus der internen Host-Liste den Maximalwert erreicht und steht dieser Host auch in der Eingabe-Host-Liste, wird der Eintrag aus der Eingabe-Host-Liste entfernt. Die verschiedenen Zähler für die Returncodes werden aktualisiert.

Nachfolgendes Beispiel zeigt den Exit-Code für die oben beschriebene Bearbeitung:

```
* *****
*
*      COPYRIGHT (C) SIEMENS AG 1990
*              ALL RIGHTS RESERVED
*
* *****
*  MODULE-DESCRIPTION
*
*  TITLE          (/ EXIT#097 ASC MODULE /)
*
*  NAME           EXIT#097
*
*  DOMAIN        SPOOL
*
*  LANGUAGE       ASC
*
*  COPYRIGHT (C) SIEMENS AG 1993
*              ALL RIGHTS RESERVED
*
*  MEMORY-CLASS  4
*
*  PROG-CONVENTS (/ ASS columbus conventions /)
*
*
*  PROCEDURE      BLDHSTL
*  REMARKS        (/ build host list table /)
*  ENTRY          BLDHSTL      EXTERNAL
*
*  DATA          DCHDESC
*  REMARKS        (/ chain descriptor /)
*  ENTRY          CHDESC      INTERNAL
*
*  DATA          SSVSVT
```

```

*      REMARKS      (/ SPOOL table /)
*      ENTRY        SVT          EXTERNAL
*
*
*      DATA        EXTERNAL_GC_USED
*      REMARKS      (/ none /)
*      ENTRY        EXTERNAL_GC_USED  EXTERNAL
*
*
*      END-MODULE-DESCRIPTION EXIT#097.
*****
*
*      LEV0:  EXIT#097
*              !
*              +-----+-----+
*              !         !         !
*      LEV1:  CHKHST      !         !
*              !         !         !
*      LEV2:  BLDHST      !         !
*              !         !         !
*      LEV3:  ADD         !         !
*              !         !         !
*      LEV4:  SEARCH      LOCK      UNLOCK
*
*****
*
*      LEV0:  NLKRES34
*              !
*              +-----+-----+-----+-----+
*              !         !         !         !         !
*      LEV1:  SDFANAL      !         !         !         !
*              !         !         !         !         !
*      LEV3:  SHOW         ADD         MODIFY      REMOVE
*              !         !         !         !         !
*              +-----+-----+-----+-----+
*              !         !         !         !         !
*      LEV4:  SEARCH      LOCK      UNLOCK
*
*****
*
EXIT#097 START
EXIT#097 RMODE ANY
EXIT#097 AMODE ANY
##BAL OPSYN ##BAS
##BALR OPSYN ##BASR
*
*      ENTRY EXIT#097
EXIT#097 @ENTR TYP=I,LOCAL=STMWA,TITLE=NO,ENV=SPLSPEC
        @DATA CLASS=B,BASE=R11,DSECT=CLREXPL

```

```

*
      LR      R11,R1
      @PASS  NAME=LOCK
      @PASS  NAME=CHKHST
      XC     E97RET,E97RET
      @PASS  NAME=UNLOCK
      @EXIT
      @END
      EJECT
      ENTRY  NLKRES34
NLKRES34 DS    OF
NLKRS34  @ENR  TYP=I,LOCAL=STMWA,TITLE=NO,ENV=SPLSPEC
*****
* UNIT-BODY EXIT#097.NLKRES34
*
* TITLE (/
*           NLKRES34 - main routine for cmd interface
* /)
* PURPOSE
* This function is the main entry used for the support of the
* interface commands.
*
*                                           END-PURPOSE*
*
* REALIZATION
*
* CONTROL (/
*   call SDFANAL   for SDF transfer area analysis
*   call LOCK
*   case cmd.internal_name
*   = $SPSDP1 : call SHOW
*   = $SPSDP2 : call ADD
*   = $SPSDP3 : call MODIFY
*   = $SPSDP4 : call REMOVE
*   call UNLOCK
* /)
*                                           END-REALIZATION*
*
* PARAMETERS
*
*   IN   R1           (/ SDF transfer area address /)
*   OUT  NONE         (/ none /)
*                                           END-PARAMETERS*
*
* END-BODY EXIT#067.NLKRES34.
*****
*
      @PASS  NAME=SDFANAL
*

```

```

@PASS NAME=LOCK
@CAS2 CMDNAMX
@OF   =CL8'$SPSDP1 '
@PASS NAME=SHOW
@OF   =CL8'$SPSDP2 '
@PASS NAME=ADD
@OF   =CL8'$SPSDP3 '
@PASS NAME=MODIFY
@OF   =CL8'$SPSDP4 '
@PASS NAME=REMOVE
@BEND
@PASS NAME=UNLOCK
@EXIT
@END
EJECT
TITLE 'WORK AREA'

```

\*\*\*\*\*

```
STMWA  @PAR  D=YES
```

\*

```
LEV1   DS    16F
```

```
LEV2   DS    16F
```

```
LEV3   DS    16F
```

```
LEV4   DS    16F
```

\*

```
JOBSTAT DS    X
```

```
JOBWFT  EQU  X'01'
```

```
JOBFT   EQU  X'02'
```

\*

300

```
SEARCHH DS    CL8
```

\*

```
LIMDEC  DS    XL3
```

```
NBRDEC  DS    XL5
```

```
PACKED  DS    D
```

\*

```
WROUTPL WROUT MF=C,PARMOD=31
```

\*

\*\*\*\*\*

\* Parameter list created by SDFANAL routine

\* Input for SEARCH,ADD,MODIFY and REMOVE routines

\*

```
CMDPL   DS    OF      Transfer area result and pl for subroutines
```

```
CMDNAMX DS    CL8     command internal name
```

```
CMDHOST DS    CL8     host name operand
```

```
CMDLIM  DS    H       limit value
```

\*\*\*\*\*

\* Output from SEARCH routine

\*

```
SEARCHRC DS    X      RC from SEARCH routine
```

```
not_found EQU  X'01'  - entry not found
```

```
found     EQU  X'00'  - entry found
```

```

*
ENTRY@  DS   A           Address returned by SEARCH routine
*****
* Input for SENDMSG routine
*
MSG#    DS   X           id of the msg sent to sysout
already_defined EQU  X'00'
not_defined   EQU  X'01'
table_full    EQU  X'02'
no_entry      EQU  X'03'
*
DISPBUF DS   0F
DISPHDR DS   H
DISPRES DS   3X
DISPAREA DS  0X
*
DISPREC DS   0X
DISPNAME DS  CL8
DISPTXT DS  CL3
DISPLIM DS  CL3
DISPTXT2 DS  CL8
DISPWFT# DS  CL4
DISPTXT3 DS  CL7
DISPFT# DS  CL4
DISPCR DS   X
DISPLEN EQU  *-DISPREC
*
          ORG   DISPREC
DISPTXT0 DS  CL16
DISPGLIM DS  CL3
DISPLEN1 EQU  *-DISPREC
          ORG   ,
*
          PRINT NOGEN, CODE
STMWA    @PAR  LEND=YES
          EJECT
CHKHST   @ENTR TYP=L, BASE=R10, TITLE=NO
          @DATA CLASS=B, BASE=R11, DSECT=CLREXPL
          @DATA CLASS=B, BASE=R13, DSECT=STMWA
*
          STM   R14, R12, LEV1
          LR   R10, R15
          XC   E97RET, E97RET
*
          L    R2, E97SVT@
          @PASS NAME=BLDHSTL
*
          @DATA CLASS=B, BASE=R5, DSECT=HOSTREC
          L    R3, E97HSTL@
          LH   R4, E97HST#

```

```

XR   R6,R6
XR   R7,R7
XR   R8,R8
*
@CYCL ,
LA   R5,HOSTTBL
*
MVC  CMDHOST,0(R3)
@PASS NAME=SEARCH
*
@IF  EQ
CLI  SEARCHRC,found
@THEN
L    R5,ENTRY@
@IF  GE
CLC  HOSTFT#,HOSTLIM
@THEN
XC   0(8,R3),0(R3)
@ELSE
AH   R6,HOSTWFT#
AH   R7,HOSTFT#
LA   R8,1(,R8)
@BEND
@BEND
*
AH   R3,E97HSTL
BCTR R4,0
@WHEN ZE
LTR  R4,R4
@BREA
@BEND
*
STH  R6,E97WFT#
STH  R7,E97FT#
STH  R8,E97HOK#
*
LM   R14,R12,LEV1
@EXIT
@END
EJECT
BLDHSTL @ENTR TYP=L,BASE=R10,TITLE=NO
@DATA CLASS=B,BASE=R2,DSECT=SVT
@DATA CLASS=B,BASE=R3,DSECT=CHDESC
@DATA CLASS=B,BASE=R4,DSECT=SSL0T
@DATA CLASS=B,BASE=R5,DSECT=HOSTREC
@DATA CLASS=B,BASE=R13,DSECT=STMWA
*
STM  R14,R12,LEV2
LR   R10,R15
*
```

```

        DROP R15
        LA   R15,HOSTTBL
        @DATA CLASS=B,BASE=R15,DSECT=HOSTREC
*      Clear counters
        @CYCL ,
        XC   HOSTWFT#(4),HOSTWFT#
        LA   R15,HOSTLG(,R15)
        @WHEN EQ
        CLC  HOSTNAME,=C'*END      '
        @BREA
        @BEND
*
        DROP R15
*
        L    R3,SVTUNS1+2
        L    R4,CHDTABA
        @WHIL NZ
        LTR  R4,R4
        @DO
        XC   JOBSTAT,JOBSTAT
        @IF  MI
        TM   SSLJBTY,SSLJBDP+SSLCPSV
        @THEN
        @IF  EQ
        CLI  SSLSTAT,SSLFT
        @THEN
        MVI  JOBSTAT,JOBFT
        @ELSE
        @IF  EQ
        CLI  SSLSTAT,SSLWFT
        @THEN
        @IF  EQ
        CLI  SSLFTST,SSLFTRQ
        @THEN
        MVI  JOBSTAT,JOBFT
        @ELSE
        MVI  JOBSTAT,JOBWFT
        @BEND
        @BEND
        @BEND
        @BEND
*
        @IF  NZ
        CLI  JOBSTAT,0
        @THEN
        @IF  EQ
        CLI  SSLCLTT,SSLCLHO
        @THEN
        MVC  SEARCHH,SSLOHST
        @ELSE

```

```

MVC  SEARCHH,SSLIHST
    @BEND
*
MVC  CMDHOST,SEARCHH
    @PASS NAME=SEARCH
*
    @IF  EQ
    CLI  SEARCHRC,not_found
    @THEN
    @IF  NZ
    OC   ENTRY@,ENTRY@
    @THEN
    L    R5,ENTRY@
    MVC  HOSTNAME,SEARCHH
    MVC  HOSTLIM,FTLIMIT
    @BEND
    @ELSE
    L    R5,ENTRY@
    @BEND
*
    @IF  EQ
    CLI  JOBSTAT,JOBWFT
    @THEN
    LH   R6,HOSTWFT#
    LA   R6,1(,R6)
    STH  R6,HOSTWFT#
    @ELSE
    LH   R6,HOSTFT#
    LA   R6,1(,R6)
    STH  R6,HOSTFT#
    @BEND
    @BEND
    LH   R6,CHDPTRS
    L    R4,4(R6,R4)
    @BEND
    LM   R14,R12,LEV2
    @EXIT
    @END
    EJECT
SDFANAL @ENTR TYP=L,BASE=R10,TITLE=NO
        @DATA CLASS=B,BASE=R1,DSECT=C$CMDINT
        @DATA CLASS=B,BASE=R2,DSECT=CMDHEAD
        @DATA CLASS=B,BASE=R3,DSECT=CMDODES
        @DATA CLASS=B,BASE=R4,DSECT=CMDOVAL
        @DATA CLASS=B,BASE=R13,DSECT=STMWA
*
    STM  R14,R12,LEV1
    LR   R10,R15
*
    DROP R15

```



```

XC    CMDLRET(4),CMDLRET          Set rc OK
XC    CMDLCRC(2),CMDLCRC         Set command rc OK
MVC   CMDLCRCM(7),=C'CMD0001'    Return msg CMD0001
LA    R1,CMDLTA                  Get SDF transfer area address
@DATA CLASS=B,BASE=R1,DSECT=CMDSDES
LA    R2,CMDMAINO                Get main operand address
LR    R3,R2

*
MVC   CMDNAMX,CMDINTN           internal name of the command
*
*****
*      Host name operand          *
*****
@IF   GE
CLC   CMDNRMO,=AL2(1)
@THEN ,
@IF   ON
TM    CMDGSTAT,CMDOCC
@THEN
ICM   R4,15,CMDOPTR
MVC   CMDHOST,CMDAVAL           move input host name
@ELSE
XC    CMDHOST,CMDHOST
@BEND
@BEND
*
*****
*      Limit operand             *
*****
@IF   EQ
CLC   CMDNRMO,=AL2(2)
@THEN ,
LA    R2,CMDHEADL(,R2)
LR    R3,R2
@IF   ON
TM    CMDGSTAT,CMDOCC
@THEN
ICM   R4,15,CMDOPTR
XC    CMDLIM,CMDLIM
MVC   CMDLIM+1(1),CMDAVAL      move input limit
@BEND
@BEND
LM    R14,R12,LEV1
@EXIT
@END
EJECT
SHOW  @ENTR TYP=L,BASE=R10,TITLE=NO
@DATA CLASS=B,BASE=R3,DSECT=HOSTREC
@DATA CLASS=B,BASE=R13,DSECT=STMWA
*

```

```

        STM   R14,R12,LEV3
        LR    R10,R15
*
        DROP R15
        XR    R5,R5
*
* Display first line : 'DEFAULT LIMIT : @@@'
*
        MVC   DISPTXT0,=C'DEFAULT LIMIT : '
        XR    R1,R1
        LH    R1,FTLIMIT
        CVD   R1,PACKED
        UNPK  LIMDEC(3),PACKED+6(2)
        OI    LIMDEC+2,X'FO'
        MVC   DISPGLIM,LIMDEC
*
        LA    R4,DISPLEN1+5
        STH   R4,DISPHDR
        XC    DISPRES,DISPRES
        MVC   WROUTPL(CUWL@WRO),WROUTL
        LA    R15,DISPBUF
        ST    R15,CUWMSGW
        LA    R15,ERR001
        ST    R15,CUWERRW
        WROUT PARMOD=31,MF=(E,WROUTPL)
*
* Display one line per host in the table
* 'hostname : @@@ WFT# : @@@ FT# : @@@'
*
        LA    R3,HOSTTBL
*
        @CYCL ,
        @WHEN EQ
        CLC   HOSTNAME,=C'*END      '
        @BREA
        @IF   NZ
        OC    HOSTNAME,HOSTNAME
        @THEN
        LA    R5,1(,R5)
        MVC   DISPNAME,HOSTNAME
        MVC   DISPTXT,=C' : '
        XR    R1,R1
        LH    R1,HOSTLIM
        CVD   R1,PACKED
        UNPK  LIMDEC(3),PACKED+6(2)
        OI    LIMDEC+2,X'FO'
        MVC   DISPLIM,LIMDEC
*
        MVC   DISPTXT2,=C' WFT# : '
*

```

```

LH    R1,HOSTWFT#
CVD   R1,PACKED
UNPK  NBRDEC(5),PACKED+5(3)
OI    NBRDEC+4,X'FO'
MVC   DISPWFT#(4),NBRDEC+1
*
MVC   DISPTXT3,=C' FT# : '
*
LH    R1,HOSTFT#
CVD   R1,PACKED
UNPK  NBRDEC(5),PACKED+5(3)
OI    NBRDEC+4,X'FO'
MVC   DISPFT#(4),NBRDEC+1
*
MVI   DISPCR,CHRNL
LA    R4,DISPLEN+5
STH   R4,DISPHDR
XC    DISPRES,DISPRES
MVC   WROUTPL(CUWL@WRO),WROUTL
LA    R15,DISPBUF
ST    R15,CUWMSGW
LA    R15,ERR001
ST    R15,CUWERRW
WROUT PARMOD=31,MF=(E,WROUTPL)
@BEND
LA    R3,HOSTLGT(,R3)
@BEND
*
* No entry in the table => send error message
*
@IF   ZE
LTR   R5,R5
@THEN
MVI   MSG#,no_entry
@PASS NAME=SENDMSG
@BEND
*
ERR001 DS   0H
LM    R14,R12,LEV3
@EXIT
@END
EJECT
ADD   @ENTR TYP=L,BASE=R10,TITLE=NO
@DATA CLASS=B,BASE=R3,DSECT=HOSTREC
@DATA CLASS=B,BASE=R13,DSECT=STMWA
*
STM   R14,R12,LEV3
LR    R10,R15
*
DROP R15

```

```

*
* Search host name in the table
* if found then send error message
* if not found and not space in the table then send error message
* if not found and free space then create entry in the table
*
      @PASS NAME=SEARCH
      @CAS2 SEARCHRC,COMP=CLI
      @OF (found)
      MVI MSG#,already_defined
      @PASS NAME=SENDMSG
      @OF (not_found)
      @IF ZE
      OC ENTRY@,ENTRY@
      @THEN
      MVI MSG#,table_full
      @PASS NAME=SENDMSG
      @ELSE
      L R3,ENTRY@
      MVC HOSTNAME,CMDHOST
      MVC HOSTLIM,CMDLIM
      @BEND
      @BEND
      LM R14,R12,LEV3
      @EXIT
      @END
      EJECT
MODIFY @ENTR TYP=L,BASE=R10,TITLE=NO
      @DATA CLASS=B,BASE=R3,DSECT=HOSTREC
      @DATA CLASS=B,BASE=R13,DSECT=STMWA
*
      STM R14,R12,LEV3
      LR R10,R15
*
      DROP R15
*
* If hostname not given (*NONE in cmd)
* then update default limit and return
*
      @IF ZE
      OC CMDHOST,CMDHOST
      @THEN
      MVC FTLIMIT,CMDLIM
      LM R14,R12,LEV3
      @EXIT
      @BEND
*
* Search host name in the table
* if not found then send error message
* if found then modify the limit

```

```

*
@PASS NAME=SEARCH
@CAS2 SEARCHRC,COMP=CLI
@OF (not_found)
MVI MSG#,not_defined
@PASS NAME=SENDMSG
@OF (found)
L R3,ENTRY@
MVC HOSTLIM,CMDLIM
@BEND
LM R14,R12,LEV3
@EXIT
@END
EJECT
REMOVE @ENTR TYP=L,BASE=R10,TITLE=NO
@DATA CLASS=B,BASE=R3,DSECT=HOSTREC
@DATA CLASS=B,BASE=R13,DSECT=STMWA
*
STM R14,R12,LEV3
LR R10,R15
*
DROP R15
*
* Search host name in the table
* if not found then send error message
* if found then remove the entry in the host table
*
@PASS NAME=SEARCH
@CAS2 SEARCHRC,COMP=CLI
@OF (not_found)
MVI MSG#,not_defined
@PASS NAME=SENDMSG
@OF (found)
L R3,ENTRY@
XC HOSTREC(HOSTLGT),HOSTREC
@BEND
LM R14,R12,LEV3
@EXIT
@END
EJECT
SEARCH @ENTR TYP=L,BASE=R10,TITLE=NO
@DATA CLASS=B,BASE=R13,DSECT=STMWA
@DATA CLASS=B,BASE=R3,DSECT=HOSTREC
*
STM R14,R12,LEV4
LR R10,R15
*
DROP R15
*
* search input host name in the table

```

```

* if found return its address in ENTRY@
* if not found then return rc + first free entry in the table
* return ENTRY@ null if no free entry
*
    LA    R3,HOSTTBL
    XR    R6,R6
    @CYCL ,
    MVI   SEARCHRC,not_found
    XC    ENTRY@,ENTRY@
    @WHEN EQ
    CLC   HOSTNAME,=C'*END    '
    @BREA
*
    ST    R3,ENTRY@
    MVI   SEARCHRC,found
    @WHEN EQ
    CLC   HOSTNAME,CMDHOST
    @BREA
*
    MVI   SEARCHRC,not_found
    @IF   ZE
    LTR   R6,R6
    @AND  EQ
    OC    HOSTNAME,HOSTNAME
    @THEN
    LR    R6,R3
    @BEND
*
    LA    R3,HOSTLGT(,R3)
    @BEND
*
    @IF   EQ
    CLI   SEARCHRC,not_found
    @AND  NZ
    LTR   R6,R6
    @THEN
    ST    R6,ENTRY@
    @BEND
    LM    R14,R12,LEV4
    @EXIT
    @END
    EJECT
LOCK    @ENTR TYP=L,BASE=R10,TITLE=NO
        @DATA CLASS=B,BASE=R13,DSECT=STMWA
*
    STM   R14,R12,LEV4
    LR    R10,R15
*
    DROP R15
    LA    R3,1

```

```

LA      R8,10
@CYCL (R8),
XR      R2,R2
@WHEN  EQ
CS      R2,R3,LOCKZ
@BREA
VPASS  1
@BEND

*

LM      R14,R12,LEV4
@EXIT
@END
EJECT
UNLOCK @ENR  TYP=L,BASE=R15,TITLE=NO
        @DATA CLASS=B,BASE=R13,DSECT=STMWA

*

XC      LOCKZ,LOCKZ

*

@EXIT
@END
EJECT
SENDMSG @ENR  TYP=L,BASE=R10,TITLE=NO
        @DATA CLASS=B,BASE=R13,DSECT=STMWA

*

STM     R14,R12,LEV4
LR      R10,R15

*

DROP  R15
@CAS2 MSG#,COMP=CLI
@OF   (already_defined)
WROUT MSG001,ERROUT,PARMOD=31
@OF   (not_defined)
WROUT MSG002,ERROUT,PARMOD=31
@OF   (table_full)
WROUT MSG003,ERROUT,PARMOD=31
@OF   (no_entry)
WROUT MSG004,ERROUT,PARMOD=31
@BEND

*
ERROUT DS    0H
        LM    R14,R12,LEV4
        @EXIT
        @END
        TITLE 'STATIC AREA'
WROUTL WROUT MSG001,MSG002,MF=L,PARMOD=31

*
* OUTPUT MESSAGES
*
MSG001 DC    Y(MSG001L)
        DS    3X

```

```

      DC      'ERROR: HOST NAME ALREADY DEFINED'
MSG001L EQU  *MSG001
*
MSG002  DC    Y(MSG002L)
      DS    3X
      DC    'ERROR: HOST NAME NOT DEFINED'
MSG002L EQU  *MSG002
*
MSG003  DC    Y(MSG003L)
      DS    3X
      DC    'ERROR: HOST TABLE FULL'
MSG003L EQU  *MSG003
*
MSG004  DC    Y(MSG004L)
      DS    3X
      DC    'NO HOST DEFINED IN THE TABLE'
MSG004L EQU  *MSG004
*
FTLIMIT DC    Y(10)
LOCKZ   DC    A(0)
      DS    0F
HOSTTBL DS    XL4096
HOSTEND DC    '*END      '
      TITLE 'DSECT AREA'
HOSTREC DSECT
HOSTNAME DS    CL8
HOSTWFT# DS    H
HOSTFT#  DS    H
HOSTLIM  DS    H
HOSTRES  DS    H
HOSTLGT  EQU  *HOSTREC
*
CLREXPL EX097 D
CHDESC  DCHDESC D
      SSVSVT D
      DSLOT D
      $CMDINT MF=D
      CMDSTRU2
      VTCSET CHR
*
      TITLE 'EXIT#097- ETPND AND CROSS REFERENCE'
ETPND  EXIT#097,COMPNR=00000000,PATCH=200,CSECT=YES,
      DOMAIN=SPOOL,
      VER=002,DATE=960426
      END

```



## 7.9 Unterstützende Kommandos für den Exit 097

Der Inhalt der internen Host-Tabelle kann mit den folgenden Kommandos eingesehen und verändert werden.

### SHOW-DPRINT-FT-LIMIT

**Funktionsbereich:** SPOOL-PRINT-ADMINISTRATION, SPOOL-PRINT-SERVICES

**Anwendergruppe:** Privilegierter Benutzer

**Privilegien:** TSOS, OPERATING, PRINT-SERVICE-ADMINISTRATION

Dieses Kommando zeigt den Inhalt der internen Host-Tabelle an. In der ersten Zeile zeigt SHOW-DPRINT-FT-LIMIT den Vorgabewert für FT-Limit an, der neuen Rechnern zugewiesen wird, die implizit in die Tabelle aufgenommen wurden.

Danach gibt SHOW-DPRINT-FT-LIMIT für jeden Eintrag in der Tabelle eine Zeile aus, die die folgende Information enthält:

- den Rechnernamen,
- den Wert für FT-Limit, der diesen Rechner zugewiesen ist,
- die Anzahl der Druckaufträge, die auf den File Transfer warten, und
- die Anzahl der Druckaufträge, die gerade übertragen werden.

### Format

SHOW-DPRINT-FT-LIMIT
----------------------

### *Hinweise*

- Die Information in der Tabelle wird jedes Mal auf den neuesten Stand gebracht, wenn Exit 097 ausgeführt wird, also etwa alle 30 Sekunden.
- Die Zahlenwerte unterscheiden sich von den Werten, die mit dem Kommando SHOW-PRINT-JOB-STATUS abgefragt werden können. Ursache dafür ist die auf Sicherheit bedachte Einstufung des Auftragsstatus (siehe Abschnitt „Exit-Verarbeitung“ auf [Seite 573](#)).

## MODIFY-DPRINT-FT-LIMIT

**Funktionsbereich:** SPOOL-PRINT-ADMINISTRATION, SPOOL-PRINT-SERVICES

**Anwendergruppe:** Privilegierter Benutzer

**Privilegien:** TSOS, OPERATING, PRINT-SERVICE-ADMINISTRATION

Mit diesem Kommando kann der Vorgabewert für FT-Limit oder der aktuelle Wert für FT-Limit eines Rechners geändert werden.

### Format

<b>MODIFY-DPRINT-FT-LIMIT</b>
<b>HOSTNAME = *NONE</b> / <alphanum_1..8> , <b>LIMIT =</b> <integer_0..255>

### Operandenbeschreibung

**HOSTNAME=\*NONE** / <alphanum\_1..8>

Mit diesem Operanden wird festgelegt, ob der angegebene Wert für FT-Limit nur für einen einzelnen Rechner oder als Vorgabe für alle Rechner gelten soll, die noch nicht in der Host-Tabelle eingetragen sind.

**HOSTNAME=\*NONE**

Der angegebene Wert für FT-Limit gilt als Vorgabe für alle noch nicht in der Tabelle eingetragenen Rechner.

**HOSTNAME=<alphanum\_1..8>**

Der angegebene Wert für FT-Limit gilt ab jetzt für den hier angegebenen Rechner.

**LIMIT=<integer\_0..255>**

Mit diesem Operanden wird der Wert für FT-Limit angegeben.

## ADD-DPRINT-FT-LIMIT

**Funktionsbereich:** SPOOL-PRINT-ADMINISTRATION, SPOOL-PRINT-SERVICES

**Anwendergruppe:** Privilegierter Benutzer

**Privilegien:** TSOS, OPERATING, PRINT-SERVICE-ADMINISTRATION

Mit diesem Kommando wird ein neuer Rechner in die Host-Tabelle eingetragen. Der für diesen Rechner geltende Wert für FT-Limit wird zusammen mit dem Rechnernamen angegeben.

### Format

<b>ADD-DPRINT-FT-LIMIT</b>
<b>HOSTNAME</b> = <alphanum_1..8> , <b>LIMIT</b> = <integer_0..255>

### Operandenbeschreibung

**HOSTNAME=<alphanum\_1..8>**

Der hier angegebene Rechner wird in die Host-Tabelle mit dem unter LIMIT angegebenen Wert für FT-Limit eingetragen.

**LIMIT=<integer\_0..255>**

Mit diesem Operanden wird der Wert für FT-Limit angegeben.

## REMOVE-DPRINT-FT-LIMIT

**Funktionsbereich:** SPOOL-PRINT-ADMINISTRATION, SPOOL-PRINT-SERVICES

**Anwendergruppe:** Privilegierter Benutzer

**Privilegien:** TSOS, OPERATING, PRINT-SERVICE-ADMINISTRATION

Mit diesem Kommando kann der angegebene Rechner aus der Host-Tabelle entfernt werden.

### Format

<b>REMOVE-DPRINT-FT-LIMIT</b>
<b>HOSTNAME</b> = <alphanum_1..8>

### Operandenbeschreibung

**HOSTNAME=<alphanum\_1..8>**

Der angegebenen Rechner wird aus Host-Tabelle entfernt.

---

# 8 Anhang

## 8.1 Makrosyntax

### Makroaufrufformat

Das Makroaufrufformat ist aus zwei Spalten aufgebaut; die erste Spalte enthält den Makronamen, die zweite Spalte die möglichen Operanden.

Makroname	Operanden
<makroname>	<operand <sub>1</sub> > ,<operand <sub>2</sub> >

Beim Aufruf des Makros muss der Makroname mit mindestens einer Leerspalte vom ersten Operanden getrennt sein. Mehrere Operanden müssen durch Kommata getrennt angegeben werden.

Im Makroaufrufformat werden bestimmte Zeichen (Metazeichen) verwendet, die in den folgenden Tabellen erläutert werden.

## „Neue“ Metasyntax der Makroaufrufformate

### Elemente der Metasyntax

Kennzeichnung	Bedeutung	Beispiele
GROSSBUCHSTABEN	Großbuchstaben bezeichnen Schlüsselwörter oder Konstanten, die in dieser Form vom Benutzer angegeben werden müssen. Schlüsselwörter müssen mit * beginnen, falls alternativ sowohl Schlüsselwörter als auch Namen von Konstanten oder Variablen angegeben werden können.	DIB FORCED=*YES
Kleinbuchstaben	Kleinbuchstaben bezeichnen Datentypen der Werte oder Variablen, die vom Benutzer angegeben werden können.	DIB = <var: pointer>
< >	Spitze Klammern kennzeichnen Variablen, deren Wertevorrat durch die Datentypen beschrieben wird.	<var: pointer>
<u>Unterstreichung</u>	Der Unterstrich kennzeichnet den Default-Wert eines Operanden. Hat ein Operand keinen Default-Wert, so ist die Angabe eines Operanden Pflicht.	FORCED = <u>*NO</u> / *YES
=	Das Gleichheitszeichen verbindet den Operandennamen mit den dazugehörigen Operandenwerten.	DATA = <var: pointer>
/	Der Schrägstrich trennt alternative Operandenwerte.	FORCED = <u>*NO</u> / *YES
list-poss(n)	Aus den list-poss folgenden Operandenwerten kann eine Liste gebildet werden. n gibt die maximale Anzahl der Listenelemente an. Enthält die Liste mehr als ein Element, muss sie in runden Klammern eingeschlossen werden.	FLAG=list-poss(3): *SLI / *SKIP / *DC  Angabe: FLAG=*SKIP FLAG=( <u>*SLI</u> ,*DC)

Einem Operanden wird durch ein Gleichheitszeichen ein Operandenwert zugewiesen, welcher aus einem definierten Wertevorrat stammt.

Dieser Wertevorrat wird durch einen Datentyp bestimmt. Nachfolgende Tabelle enthält die Datentypen der Operandenwerte.

## Datentypen der Operandenwerte

Datentyp	Zeichenvorrat	Anmerkungen
c-string	EBCDIC-Zeichen	ist in Hochkommata einzuschließen
integer	[+-] 0..2147483647	ist eine dezimale Zahl
var:	leitet eine variable Angabe ein. Nach dem Doppelpunkt folgt der Typ der Variablen (siehe Tabelle „Datentypen der Variablen“ )	<var:var-type>
reg:	Register 0..15	Angabe: (<reg:var-type>)

## Zusätze zu Datentypen

Zusatz	Bedeutung
n..m	für Datentyp integer bedeutet n..m eine Intervallangabe; n: Mindestwert m: Maximalwert
	für Datentyp c-string bedeutet n..m eine Längenangabe in Bytes; n: Mindestlänge m: Maximallänge mit $n < m$
n	bei Datentyp c-string bedeutet n eine Längenangabe in Bytes; n muss exakt eingehalten werden.

Die Operandenwerte können direkt als Zeichenkette oder Integer-Zahl (siehe Datentypen "c-string" und "integer") eingegeben werden oder indirekt über eine Variable (siehe Datentyp "var:") bezeichnet werden. Die nachfolgende Tabelle enthält die Datentypen, die für Variablen möglich sind.

## Datentypen der Variablen

Datentyp	Beschreibung	Definition im Programm
char:n	Die Variable ist eine Zeichenkette von n Zeichen. Fehlt die Längenangabe, wird n=1 angenommen.	CLn
int:n	Die Variable ist eine Integer-Zahl, die n Bytes belegt. Fehlt die Längenangabe, wird n=1 angenommen. Bedingung: $n \leq 4$	FLn
enum-of E:n	Die Variable ist die Aufzählung E, die n Bytes belegt. Fehlt die Längenangabe, wird n=1 angenommen. ( $n \leq 4$ )	XLn
pointer	Die Variable ist eine Adresse oder ein Adresswert.	A

## „Alte“ Metasyntax der Makroaufrufformate

Bei der alten Darstellung von Formaten werden bestimmte Zeichen (so genannte Metazeichen) verwendet und Vereinbarungen getroffen, die in der folgenden Tabelle erläutert sind:

Formale Darstellung	Erläuterung	Beispiel
GROSSBUCHSTABEN	Großbuchstaben bezeichnen Konstanten, die in dieser Form vom Benutzer eingegeben werden müssen.	FSTAT ,LIST=(SYSLST)  Einzugeben ist: FSTAT ,LIST=(SYSLST)
Kleinbuchstaben	Kleinbuchstaben bezeichnen Variablen, die bei der Eingabe vom Benutzer durch aktuelle Werte ersetzt werden müssen, d.h. ihr Inhalt kann von Fall zu Fall verschieden sein.	FILE dateiname  Einzugeben ist: FILE DATEI, FILE XYZ, FILE A.B-1, usw.
{ }	Geschweifte Klammern schließen Alternativen ein, d.h. aus den eingeschlossenen Größen muss eine Angabe ausgewählt werden.	{ FILE=pfadname } { LINK=name }  einzugeben ist: FILE=... oder LINK=...

Metasyntax (Teil 1 von 2)



Formale Darstellung	Erläuterung	Beispiel
[ ]	<p>Eckige Klammern schließen Wahlangaben ein, d.h. Angaben, die man weglassen darf. Steht bei Wahlangaben das Komma innerhalb Klammer, so wird es nur bei Verwendung dieser Wahlangabe verlangt und kann beim ersten Operanden weggelassen werden. Steht es außerhalb der Klammer, so muss es stets geschrieben werden, auch wenn keine Wahlangabe gemacht wird. (Runde Klammern müssen eingegeben werden)</p>	<p>F[REE]SIZE eingegeben ist FREESIZE oder abgekürzt FSIZE</p>
—	<p>Die Unterstreichung hebt den Standardwert (Voreinstellung) hervor. Das ist der Wert, den das System einsetzt, wenn der Benutzer keine Angabe macht.</p>	<p>{ { <u>ISAM</u> } } Eingeben ist: SAM oder ISAM oder nichts (= ISAM)</p>
...	<p>Punkte bedeuten eine Wiederholung. Sie zeigen an, dass die davor stehende Einheit mehrmals hintereinander wiederholt werden kann.</p>	<p>(vsn,...) Eingeben ist: (PVT003) oder (PVT003,PVT456) oder (XY00AB,XY0012,XY0005) usw.</p>
_	<p>Dieses Zeichen kennzeichnet ein Leerzeichen (X'40')</p>	<p>STD_ Anzugeben ist 'STD '</p>

Metasyntax (Teil 2 von 2)

## Typen von Makroaufrufen

In Typen werden die Makroaufrufe abhängig von der Art ihrer Operandenübergabe eingeteilt. Es gibt den O-Typ, R-Typ (Übergabe in Registern) und den S-Typ (Übergabe im Speicher).

### O-Typ-Makroaufrufe

Makroaufrufe, die weder dem R-Typ noch dem S-Typ zuzuordnen sind, werden als O-Typ-Makros bezeichnet.

Vom O-Typ sind z.B. jene Makroaufrufe, die im Operandenfeld die Angabe eines Registers vorsehen (häufig nur R1), das die Anfangsadresse einer Operandenliste enthält.

Die Operandenliste wird im Datenteil des Programms definiert (DC-Anweisungen) und enthält die Operandenwerte.

### R-Typ-Makroaufrufe

Operation	Operanden
RTYP	$\left\{ \begin{array}{l} \text{operand1} \\ (r1) \end{array} \right\}, \left\{ \begin{array}{l} \text{operand2} \\ (r2) \end{array} \right\}$

Ein Makroaufruf ist vom Typ R, wenn alle erforderlichen Operandenwerte in die zwei für diesen Zweck verwendeten Register 0 und 1 geladen werden können. Durch einen R-Typ-Makroaufruf wird keine Operandenliste generiert.

Die Operanden können direkt angegeben werden oder in den Registern 0 und 1 enthalten sein.

Adressoperanden in R-Typ-Makroaufrufen können als explizite oder implizite Adressen geschrieben werden.

## S-Typ-Makroaufrufe

Name	Operation	Operanden
[opadr]	makro	$\left. \begin{array}{l} \text{operand1, \dots, operandn, MF= } \left\{ \begin{array}{l} \underline{S} \\ \underline{L} \\ C[, \text{PREFIX=p}], [\text{MACID=mac}] \\ (C, p) \\ D[, \text{PREFIX=p}] \\ (D, p) \\ M[, \text{PREFIX=p}], [\text{MACID=mac}] \end{array} \right\} \end{array} \right\}$ $\left. \begin{array}{l} \text{MF= } \left\{ \begin{array}{l} E \\ (E, \text{opadr}) \\ (E, (r)) \end{array} \right\} \end{array} \right\} [, \text{PARAM=adr}]$

Beim S-Typ werden die im Makroaufruf angegebenen Operandenwerte in Form eines Datenbereichs an den Funktionsbaustein übergeben. Der Datenbereich ist Teil der Makroauflösung. Er ist ein geeignet strukturierter Bereich, der die für die Übergabe der Operandenwerte notwendigen Daten- und Speicherdefinitionen (DC- und DS-Anweisungen) enthält.

Für alle Makros, die mit einer bestimmten Makroversion aufgerufen werden können (z.B. über die Operanden VERSION oder PARMOD), gilt: In allen Aufrufen, die sich durch den MF-Operanden unterscheiden (MF=L/E/D/C) muss der Versionsoperand den gleichen Wert haben.

Für MF unterscheidet man 6 Formen des Makroaufrufs: S-Form, E-Form, L-Form, D-Form, C-Form, M-Form.

*S-Form = Standardform*

MF=S ist Voreinstellung. Es werden zuerst der Befehlssteil und anschließend der Datenbereich generiert, unter Beachtung der im Makroaufruf angegebenen Operandenwerte. Der Datenbereich enthält keine Feldnamen und keine erläuternden Equates. Der Standardheader ist initialisiert.

*E-Form = Execute-Form*

	Operation	Operanden
[label]	makro	$MF=E,PARAM=\left\{ \begin{array}{l} \text{addr} \\ (r) \\ (1) \end{array} \right\}$

Mit der E-Form des MF-Operanden wird ein Systemaufruf (SVC) ausgelöst: der Inhalt einer Operandenliste (siehe L-Form) wird ausgewertet und die entsprechenden Operationen werden ausgeführt. Daher muss im "Execute"-Makroaufruf die Adresse der Operandenliste enthalten sein, entweder als symbolische Adresse (addr) oder in einem Register (r/1). Weitere Operanden werden nicht ausgewertet.

"label" ist die symbolische Adresse, die dem Makroaufruf im Assemblerprogramm zugeordnet werden kann.

*L-Form = LIST-Form*

	Operation	Operanden
label	makro	MF=L, operandenliste

Die List-Form erzeugt mit den übrigen im Makroaufruf angegebenen Operanden eine Operandenliste, allerdings ohne symbolische Adressen für die Operanden. Diese werden über die C- oder D-Form generiert. Die Adresse der Operandenliste muss beim Systemaufruf (E-Form) angegeben werden, daher kann die symbolische Adresse "label" nicht entfallen.

Die Operandenliste beginnt mit dem Standardheader (siehe [Seite 606ff](#)), dessen Felder beim Aufbau der Liste über MF=L automatisch versorgt sind. Auch wenn eine Operandenliste über die D- oder die C-Form dynamisch aufgebaut werden soll, muss sie vorher mit MF=L initialisiert werden, damit eine korrekte Versorgung des Headers gewährleistet ist.

*D-Form=DSECT-Form*

	Operation	Operanden
[label]	makro	MF=D[, PREFIX=prefix]

Die D-Form erzeugt eine DSECT für die Operandenliste des Makros. Mit dem Operanden PREFIX kann das erste Zeichen der generierten Namen geändert werden. Ist für den Makro mit "label" eine symbolische Adresse definiert, erhält die DSECT diesen Namen.

Ist "label" nicht definiert, erhält die DSECT einen makrospezifischen Standardnamen, dessen erstes Zeichen ebenfalls über PREFIX modifiziert wird. Die Operandenliste sollte vor dem DSECT-Aufruf mit der List-Form initialisiert werden, damit die korrekte Versorgung des Standardheaders gewährleistet ist.

### *C-Form*

	<b>Operation</b>	<b>Operanden</b>
[label]	makro	MF=C[, PREFIX=prefix][, MACID=id]

Wie die D-Form erzeugt die C-Form eine Operandenliste, allerdings nicht als DSECT, da keine DSECT-Anweisung generiert wird. Die Operandenliste bleibt leer, sie sollte mit einem Aufruf in L-Form initialisiert werden, damit vor allem der Standardheader korrekt versorgt ist.

Mit dem Operanden PREFIX kann das erste Zeichen der generierten Namen geändert werden, mit dem Operanden MACID auch das zweite bis vierte Zeichen (in MACID kann eine bis zu drei Zeichen lange Zeichenfolge angegeben werden). Ist der Makroaufruf mit "label" symbolisch adressiert, ist dies gleichzeitig die Adresse der Operandenliste; ist "label" nicht definiert, ist die Operandenliste nicht symbolisch adressierbar.

### *M-Form = Modifizierungs-Form*

	<b>Operation</b>	<b>Operanden</b>
[label]	makro	MF=M[, PREFIX=prefix][, MACID=id], operandenliste

Es werden Befehle (z.B. MVCs) generiert, die während des Programmlaufs in einem bereits initialisierten Datenbereich (Operandenliste) Felder mit den Operandenwerten überschreiben, die im Makroaufruf angegeben werden. Damit bietet die M-Form eine komfortable Möglichkeit, die Operandenwerte, mit denen ein Makro aufgerufen wird, dynamisch dem Programmlauf anzupassen.

Da die dafür generierten Befehle die symbolischen Adressen und Equates der C-Form bzw. D-Form benutzen, ist bei der Verwendung der M-Form sicherzustellen, dass diese Namen für die Adressierung der zu modifizierenden Operandenliste zur Verfügung stehen. Insbesondere ist darauf zu achten, dass bei einem Makroaufruf mit MF=M die ggf. Operanden PREFIX und MACID mit den gleichen Werten angegeben werden wie im zugehörigen MF=C- bzw. MF=D-Aufruf.

## Standardheader

Der Standardheader ist ein 8-Byte-Feld am Beginn der Operandenliste, das die normierte Bezeichnung der Schnittstelle enthält und Returncodes aufnehmen kann. Er wird vom jeweiligen Makro erzeugt und sollte - wenn möglich - mit der List-Form der MF-Operanden initialisiert werden.

### *Aufbau des Standardheaders*

Feldinhalt	Byte-Position	Bedeutung
UNIT	0-1	bezeichnet die Funktionseinheit, in der die gesuchte Funktion realisiert ist
FUNCTION	2	bezeichnet die Funktion (innerhalb der Funktionseinheit)
VERSION	3	bezeichnet den „Änderungsstand“, d.h. die Versionsnummer der Funktion
SUBCODE2	4	enthält den Sub-Returncode2
SUBCODE1	5	enthält den Sub-Returncode1
MAINCODE	6-7	enthält den Haupt-Returncode

### Aufbau des Standardheaders

Die Felder SUBCODE2, SUBCODE1, MAINCODE enthalten den Returncode. Der Haupt-Returncode zeigt an, ob eine Operation erfolgreich ausgeführt werden konnte. Im Fehlerfall kann durch die Sub-Returncodes der Fehler festgestellt werden.

Folgende Werte des Returncodes sind Konvention:

SUB-CODE2	SUB-CODE1	MAIN-CODE	Bedeutung
X'00'	X'00'	X'0000'	Erfolgreiche Funktionsausführung. Es gibt keine zusätzlichen Informationen zum MAINCODE.
X'01'	X'00'	X'0000'	Erfolgreiche Funktionsausführung. Es sind keine weiteren Aktionen erforderlich.
X'00'	X'01'	X'FFFF'	Die angeforderte Funktion wird nicht unterstützt (falsche Angabe für UNIT oder FUNCTION im Standardheader). Nicht behebbarer Fehler.
X'00'	X'02'	X'FFFF'	Die angeforderte Funktion ist nicht verfügbar. Nicht behebbarer Fehler.
X'00'	X'03'	X'FFFF'	Die angegebene Version der Schnittstelle wird nicht unterstützt (falsche Versionsangabe im Standardheader). Nicht behebbarer Fehler.

### Returncode-Konventionen (Teil 1 von 2)

SUB-CODE2	SUB-CODE1	MAIN-CODE	Bedeutung
X'00'	X'04'	X'FFFF'	Parameterliste ist nicht auf Wortgrenze ausgerichtet.
X'00'	X'41'	X'FFFF'	Das Subsystem ist nicht vorhanden; es muss explizit erzeugt werden.
X'00'	X'42'	X'FFFF'	Der aufrufende Ablauf ist mit dieser Schnittstelle nicht konnektiert; er muss explizit konnektiert werden.
X'00'	X'81'	X'FFFF'	Subsystem zur Zeit nicht verfügbar.
X'00'	X'82'	X'FFFF'	Subsystem im DELETE- oder HOLD-Zustand.

Returncode-Konventionen (Teil 2 von 2)

MAINCODE kennzeichnet das Ergebnis der Funktionsausführung. SUBCODE1 dient der Qualifizierung des Hauptwertes. SUBCODE2 dient der weiteren Unterteilung des Fehlers in Fehlerklassen.

Der Returncode sollte ausschließlich im Standardheader übergeben werden. In einer Übergangsphase kann der Returncode aber auch im Register R15 oder sowohl im Standardheader als auch im Register R15 übergeben werden. Um zu prüfen, ob im Standardheader ein Returncode übergeben wurde, sollte das Returncode-Feld mit X'FFFFFFFF' vorbesetzt werden. Das Ergebnis der Prüfung des Standardheaders wird in jedem Fall auch im Register R15 übergeben:

X'00000000': Standardheader richtig initialisiert; normale Ausführung.  
 X'0001FFFF': Falsche Angabe für UNIT oder FUNCTION.  
 X'0003FFFF': Falsche Angabe für VERSION.

## Musterzeichen

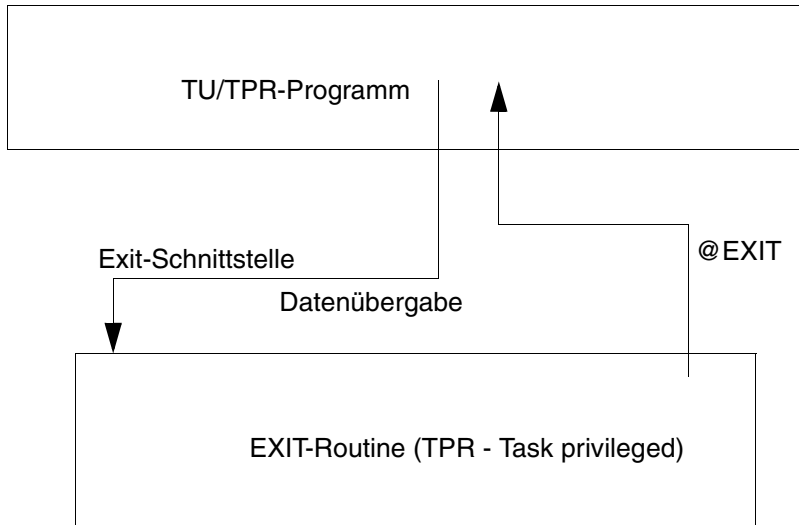
Der Anwender kann für Katalogkennung und Dateinamen Muster verwenden.

Muster	Bedeutung
*	Ersetzt eine beliebige Zeichenfolge, auch die leere Zeichenfolge.
/	Ersetzt genau ein beliebiges Zeichen.
<muster1,...>	Ersetzt alle Zeichenfolgen, auf die eines der angegebenen Muster zutrifft
<muster1:muster2>	Ersetzt eine Zeichenfolge, für die gilt: <ul style="list-style-type: none"> <li>– sie ist mindestens so lang wie die kürzeste Muster-Zeichenfolge</li> <li>– sie ist höchstens so lang wie die längste Muster-Zeichenfolge</li> <li>– sie liegt in der alphabetischen Sortierung zwischen "muster1" und "muster2"; Zahlen werden hinter Buchstaben sortiert</li> </ul> "muster1" darf auch die leere Zeichenfolge sein, die in der alphabetischen Sortierung an erster Stelle steht.
<muster1:muster2,...>	Muster der Art "muster1:muster2" können auch in Listenform angegeben werden. Für jede derartige Bereichsangabe gelten die oben genannten Regeln. Das System nimmt eine logische Oder-Verknüpfung vor, d.h. die Musterliste ersetzt alle Zeichenfolgen, auf die eine der Bereichsangaben zutrifft. Die Längenmerkmale gelten paarweise, d.h. jeweils für eine Bereichsangabe "muster1:muster2", nicht für die gesamte Liste.
-muster	Ersetzt alle Zeichenfolgen, die dem angegebenen Muster nicht entsprechen. Das Minuszeichen darf nur am Beginn der Musterzeichenfolge stehen.



## 8.2 Funktionsweise des Exit-Mechanismus

Die Funktionsweise des Exit-Mechanismus ist im folgenden Bild schematisch dargestellt:



Exit-Routinen-Aufruf

Ein System-Modul, bei dem ein Exit-Punkt vorgesehen ist, verzweigt über eine TU- bzw. TPR-Makroschnittstelle in die dazugehörige Exit-Routine, falls diese existiert und aktiv ist. Dabei übergibt der Modul exitspezifische Parameter und Daten an die Exit-Routine. Diese kann, abhängig vom gewünschten Verwendungszweck, die Daten prüfen und evtl. modifizieren, eine Anforderung abweisen oder protokollieren. Anschließend erfolgt der Rücksprung in den rufenden System-Modul durch den Makro @EXIT in Programm-Manager-Umgebung (DSL-Environment) bzw. über BR R14 bei reinen Assembler-Routinen. Neue Exitroutinen sollten unbedingt in DSL-Environment entwickelt werden.

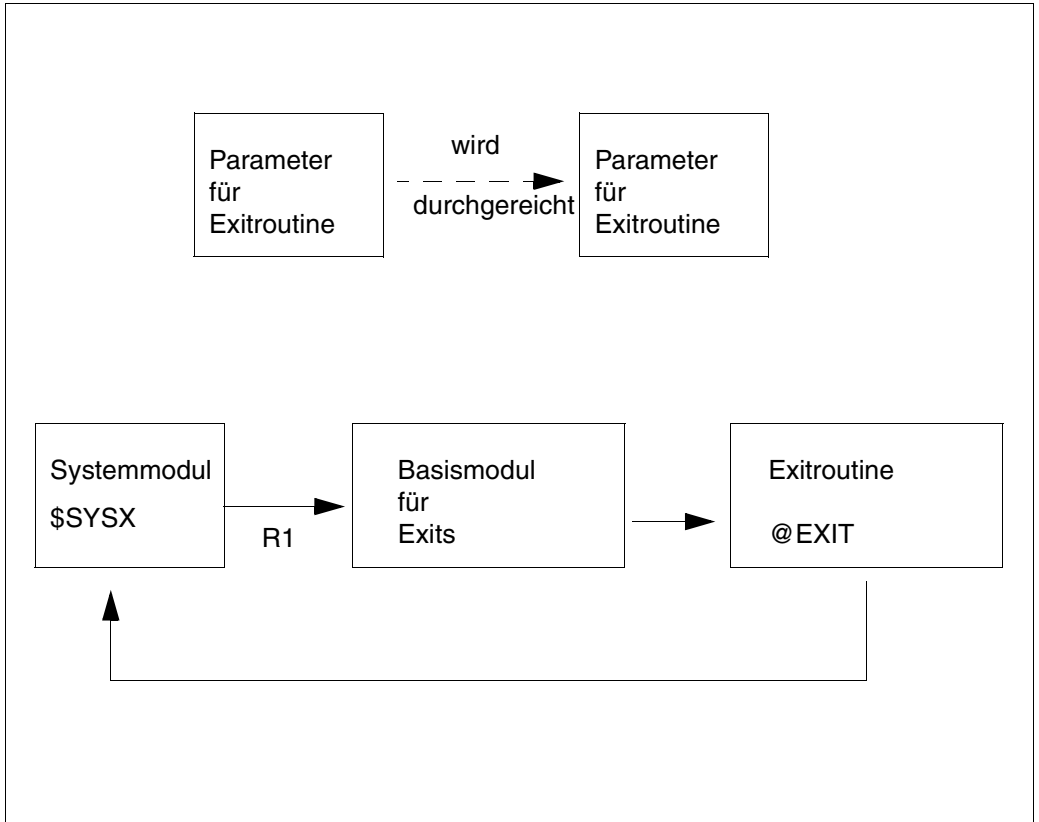
Bei der Verzweigung in eine Exit-Routine sind einige TPR-Register mit Informationen versorgt. Die Registerinhalte sind der Beschreibung des jeweiligen Exits zu entnehmen.

Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden; sie müssen sichergestellt und vor dem Rücksprung aus der Exit-Routine mit den ursprünglichen Inhalten geladen werden, d.h., sie müssen beim Rücksprung ihren ursprünglichen Inhalt haben.

Die Register 0, 2 und 3 sind nur noch für die Exits definiert, die intern eine alte System-schnittstelle verwenden. Für die Exits, die schon umgestellt wurden sind die Register 0, 2 und 3 nicht definiert. Bitte beachten Sie die Beschreibung der Exits. Weitere Informationen über Exits finden Sie im Handbuch „[System Exits \(nur in Englisch\)](#)“.

Die Zugriffe auf Inhalte der EXVT und des TCB müssen über die entkoppelten Zugriffsfunktionen erfolgen.

Die Rückkehrinformation der Exitroutine muss entsprechend der jeweiligen Exit-Beschreibung entweder in Register 15 oder in der spezifizierten Parameterliste mit Standardheader hinterlegt werden.



Exit-Mechanismus

### 8.3 Schlussblatt für lokalen SPOOL

MAIL: @@@@  
 FILE: @@@@  
 ELEM: @@@@  
 TYPE : @@@@ VERS : @@@@  
 SPOOL DATE : ####-##-## HOST : @@@@ SPOOL TSN : ####  
 START TIME : ##:##:##@ U-ID : @@@@ REQ. TSN : ####  
 END TIME : ##:##:##@ DEV. : @@@@ DEV. TYPE : @@@@

DIA :@@ (@@) FAMILY:@@ LOOP :(@@,@@) FORM :@@@@  
 STARTNO :##### ENDNO :##### FROM :#####TO :#####  
 SHIFT :### SPACE :@ BINARY:@@ CONTR:@@@@@@  
 DEL-FILE:@@@@@@ LINES :##### COPIES:(###,###) PRIOR:@@  
 HEADER :(@,@,@) HDRNUM:##### TRLNUM:#####CCPOS:#####  
 PNAME :@@@@@ TEXT :@@@@@  
 LOCK :@@@ DEST. :@@@@@ TRUNCATION:@@@@@  
 IMAGE :@@@@@  
 MONJV :@@@@@  
 TRAY :(###,###) MESSAGE=PAGE:@@ CHECKPOINT:@@@@@  
 SECT : C'@@@@@  
 POSITION : ####  
 FIRST: C'@@@@@  
 NUMBER:##### POSITION : ####  
 LAST : C'@@@@@  
 NUMBER:##### POSITION : ####  
 TR-TABLE:(@@@@@,@@@@@)  
 LANG-EXT:@@@@@  
 PAGEDEF :@@@@@ LP65DIA(###,###) DUPLEX : @@@@@  
 CHARS :(@@@,@@@,@@@,@@@) CHARPOOL : (@@@@,##)  
 FOB :@@@@ FORMDEF:@@@@@ ROTATION : #####  
 OVERLAY :( , ) OFFSET :(L: ,T: )  
 CLIENT-H:@@@@@ CLUSTER : @@@@@  
 FILE :SIZE=#####,FCBTYPE=@@@,RECFORM=@,RECSIZE=#####,BLKSIZE=#####  
 CCSNAME=@@@@@@ FILE-CR-MODE=@@@@@@  
 PRNT :#LINES=##### COPY#=##### SPOOLOUT CLASS=@@@

Layout des Schlussblattes für lokalen Spool

*Hinweise*

1. Im Feld "CLIENT-H" wird der Name des Hosts ausgegeben, auf dem der Druckauftrag abgegeben wurde.
2. Im Feld "CLUSTER" wird 'LOCAL' bei lokalen Aufträgen oder verteilten Aufträgen innerhalb des gleichen Clusters ausgegeben. 'SINIX' wird bei verteilten Aufträgen zwischen Clustern ausgegeben, wenn der Druckauftrag von einem UNIX-System-basierten Cluster gestartet wurde. 'BS2000' wird bei verteilten Aufträgen zwischen Clustern ausgegeben, wenn der Druckauftrag von einem BS2000-Cluster gestartet wurde.
3. Die Felder "START TIME" und "END TIME" wurden zur Darstellung von Sommer- und Winterzeit aktualisiert.
4. Im Feld "CCSNAME" wird Name des Coded Character Sets, der in der Datei verwendet wird, angezeigt.
5. Im Feld "FILE-CR-MODE" wird der Erstellungsmodus der Datei angezeigt, falls der Druckauftrag das Produkt DSEM angefordert hatte.

## 8.4 SPOOLOUT-Abrechnungssatz SPLO

Der SPOOLOUT-Abrechnungssatz wird bei Beendigung eines SPOOLOUT-Auftrages geschrieben.

In diesem Abrechnungssatz werden für jeden Druckauftrag Informationen über den Auftraggeber sowie den Druckauftrag und dessen Bearbeitung festgehalten. Der Abrechnungssatz enthält in jedem Fall die folgenden Daten:

- Benutzerkennung, Abrechnungsnummer und Gruppenname
- die TSN der Controller-Task und des SPOOLOUT-Jobs
- Datum und Uhrzeit mit vierstelliger Jahreszahl für Beginn und Ende der Auftragsbearbeitung

### *Hinweis*

Das Layout des Abrechnungssatzes kann mit dem Makro ACSPLO erzeugt werden.

Die bei der Darstellung pro Datenfeld verwendeten Attribute sind:

Feld	laufende Nummer des Datenfeldes innerhalb des beschriebenen Satzteils
Distanz	relativer Abstand des Datenfeldes zum Anfang des beschriebenen Satzteils
Länge	Länge des Datenfeldes in Byte
Format	A = alphanumerisch
	B = Binärzahl
	B2 = binäre Darstellung der CPU-Zeit
	C = abdruckbare Zeichen, inklusive Sonderzeichen
	F = Dateiname
	X = nicht abdruckbare Zeichen
	Z = ungepackte Dezimalzahl (*)
	* = wird bei den einzelnen Satzarten oder Erweiterungselementen festgelegt
	- = reserviert für künftige Erweiterungen und enthält entweder Leerzeichen oder binär Null
(*)	Uhrzeit wird in der Form hhhmmss dargestellt Datum wird in der Form jjmmtt dargestellt

**(A) Satzbeschreibung**

Satzkennung: 'SPLO'

**(B) Kennzeichnungsteil**

Der Kennzeichnungsteil der Abrechnungssätze für die Abrechnung von Benutzeraufträgen besteht aus einer Benutzerkennzeichnung.

Aufbau und Inhalt:

Feld-Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex	dez				
1	00	0	8	A	Benutzerkennung	
2	08	8	8	A	Abrechnungsnummer	
3	10	16	4	C	TSN	1)
4	14	20	8	A	Gruppenname	
5	1C	28	8	A	Host-Name	2)

*Anmerkung*

- 1) TSN der Controller-Task
- 2) Name des Rechners, von dem der SPOOLOUT-Auftrag angefordert wurde. Das Feld wird nur bei Einsatz des Produktes Distributed Print Services gefüllt.

Länge der Benutzerkennzeichnung: 36 Byte

**(C) Grundinformation**

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	6	Z	Datum des SPOOLOUT-Startes	1)
2	06	6	6	Z	Uhrzeit des SPOOLOUT-Startes	2)
3	0C	12	6	Z	Datum der SPOOLOUT-Beendigung	1)
4	12	18	6	Z	Uhrzeit der SPOOLOUT-Beendigung	2)
5	18	24	8	A	SPOOLOUT-Auftragsname (PNAME)	
6	20	32	4	Z	TSN des SPOOLOUT-JOBs	
7	24	36	2	B	Anzahl der noch auszugebenden Kopien	
8	26	38	1	B	SPOOLOUT-Klasse	
9	27	39	1	B	SPOOLOUT-Scheduling-Priorität	
10	28	40	3	C	Art der SPOOLOUT-Datei	3)
11	2B	43	1	-	- reserviert -	
12	2C	44	2	Z	Jahrhundert für SPOOLOUT-Start	4)
13	2E	46	1	A	Kennzeichen Jahreszeit für SPOOLOUT-Start	5)
14	2F	47	1	A	Kennzeichen Jahreszeit für SPOOLOUT-Ende	5)
15	30	48	2	Z	Jahrhundert für SPOOLOUT-Ende	4)
16	32	50	4	Z	TSN des Partner-Druckauftrages	6)

Länge der Grundinformation: 54 Byte

*Anmerkung*

1. Datum in Form yymmdd.
2. Uhrzeit in der Form hhmmss.
3. Folgende Anzeigen sind möglich:
  - 'SYS' = logische Systemdateien (SYSLST, SYSOUT, SYSOPT)
  - 'OMF' = EAM-Bindemoduldatei
  - 'EAM' = Benutzer-EAM-Datei
  - 'PLM' = PLAM-Bibliothekselement
  - ' ' = sonstige Datei
  - 'TMP' = Temporäre Datei
4. z.B. Zeichenfolge "19" als Kennzeichen für das Jahr "1993".

5. "S" für Sommerzeit; "W" für Winterzeit.
6. Zwei Fälle werden für diese TSN unterschieden:
  - Wenn der Druckauftrag auf einen fernen Rechner erteilt wurde und auf dem lokalen Server verarbeitet wird, dann ist dies die TSN auf den Client- oder Gateway-Rechner.
  - Wenn der Druckauftrag auf den lokalen Rechner erteilt wurde und auf dem lokalen Server verarbeitet wird, dann ist dies die gleiche TSN wie die des Druckauftrags.

#### (D) Variable Information

Die variable Information des SPOOLOUT-Abrechnungssatzes enthält sechs Satzerweiterungen mit folgendem Aufbau und Inhalt:

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	B	X'0007'	1)
2	02	2	2	B	Distanz der 1. Erweiterung	2)
3	04	4	2	B	Distanz der 1. bis zur 2. Erweiterung	3)
4	06	6	2	B	Distanz der 1. bis zur 3. Erweiterung	
5	08	8	2	B	Distanz der 1. bis zur 4. Erweiterung	
6	0A	10	2	B	Distanz der 1. bis zur 5. Erweiterung	
7	0C	12	2	B	Distanz der 1. bis zur 6. Erweiterung	
8	0E	14	2	B	Distanz der 1. bis zur 7. Erweiterung	
9	10	16	2	B	Distanz der 1. bis zur 8. Erweiterung	

1. Steht auf 0, wenn keine Satzerweiterung verfügbar ist.
2. Die Distanz bezieht sich auf den Beginn des Erweiterungssatzes.
3. Wird eine Erweiterung nicht unterstützt, erhält sie die Distanz 0.

Länge des Erweiterungskopfes: 18 Byte



## 1. Erweiterung: SPOOLOUT-Beendigungs-Grund

Diese Satzerweiterung ist eine Strukturfeld-Erweiterung mit einem Element.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'OT'	
2	02	2	1	B	Anzahl der Elemente (X'01')	
3	03	3	1	B	Länge des Elementes (X'0C')	
4	04	4	2	A	Beendigungs-Anzeige 'T' = normale Beendigung 'A' = abnormale Beendigung	
5	06	6	1	-	- reserviert -	
6	07	7	1	A	Beendigungs-Anforderung	1)
7	08	8	7	A/C	Beendigungscode	1)
8	0F	15	1	-	- reserviert -	

Länge der SPOOLOUT-Beendigungs-Erweiterung: 16 Byte

### Anmerkung

Beendigungsanforderung	Beendigungscode
'F'	NORM = normale Beendigung
'E'	DMS = Fehler beim Lesen der Datei
'\$'	I/O = Ein-/Ausgabefehler am Ausgabegerät
'X'	CAN = Beendigung durch /CANCEL-PRINT-JOB

## 2. Erweiterung: SPOOLOUT-Erzeugung

Diese Satzerweiterung ist eine Strukturfeld-Erweiterung mit einem Element.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'OC'	
2	02	2	1	B	Anzahl der Elemente (X'01')	
3	03	3	1	B	Länge des Elementes (X'1D')	
4	04	4	4	Z	TSN der erzeugenden Task	
5	08	8	2	-	- reserviert -	
6	0A	10	12	Z	Datum und Uhrzeit der Erzeugung des SPOOLOUT-Auftrags	
7	18	22	2	Z	Jahrhundert des Datums	
8	1A	24	8	A	Ursprüngliche Benutzerkennung (bei Replay-Jobs)	
9	20	32	1	A	Kennzeichen Jahreszeit für SPOOLOUT-Start	

Länge der SPOOLOUT-Erzeugungs-Erweiterung: 33 Byte

## 3. Erweiterung: SPOOLOUT-Anstoß

Diese Satzerweiterung ist eine Fallunterscheidungs-Erweiterung.

Sie ist nur dann vorhanden, wenn der SPOOLOUT-Vorgang durch das Kommando HOLD-PRINT-JOB (Operand RESUME-CONDITION) angestoßen wurde.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'OI'	
2	02	2	1	B	Anzahl der Elemente (X'01')	
3	03	3	1	B	Länge des Elementes (X'13')	
4	04	4	2	A	Fallkennzeichnung 'RE'	
5	06	6	2	-	- reserviert -	
6	08	8	12	-	- reserviert -	
7	14	20	2	Z	- reserviert -	
8	16	22	1	A	- reserviert -	

Länge der HOLD-PRINT-JOB-Kommando-Erweiterung: 23 Byte

#### 4. Erweiterung: SPOOLOUT-Eingabeband

Diese Satzerweiterung ist eine Strukturfeld-Erweiterung mit einem Element.

Sie wird nur dann erzeugt, wenn die ausgegebene Datei von einem Benutzer-Band oder von einem Replay-Band gelesen wurde.

Ein Benutzer-Band wird mit dem Kommando WRITE-SPOOL-TAPE erzeugt.

Ein Replay-Band ist ein Band, das mit dem Kommando START-TAPE-OUTPUT von der Systembetreuung für SPOOLOUT/SPOOLIN zugewiesen wird.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'IN'	
2	02	2	1	B	Anzahl der Elemente (X'01')	
3	03	3	1	B	Länge des Elementes (X'14')	
4	04	4	2	-	- reserviert -	
5	06	6	2	A	mnemotechnischer Name des Bandgerätes	
6	08	8	8	-	- reserviert -	
7	10	16	6	-	- reserviert -	
8	16	22	2	-	- reserviert -	

Länge der SPOOLOUT-Eingabeband-Erweiterung: 24 Byte

## 5. Erweiterung: Ausgabemedium

Diese Satzerweiterung ist eine Fallunterscheidungs-Erweiterung.

Ihre Erweiterungskennung ist in jedem Fall 'OM'.

Struktur und Inhalt hängen von der Fallkennzeichnung ab, die gleichzeitig anzeigt, auf welche Weise die SPOOLOUT-Ausgabe erfolgt ist.

### a) Zeilendrucker-Erweiterung

Die SPOOLOUT-Datei wurde auf einem SPOOL- oder RSO-Zeilendrucker ausgedruckt.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'OM'	
2	02	2	1	B	Anzahl der Elemente (X'01')	
3	03	3	1	B	Länge des Elementes (X'20')	
4	04	4	2	A	Fallkennzeichnung ' '	
5	06	6	2	A	mnemotechnischer Gerätenamen des Druckers	
6	08	8	4	B	Anzahl der gedruckten Zeilen	1)
7	0C	12	4	B	Anzahl der gedruckten Seiten	1)
8	10	16	8	A	Gerätenamen	
9	18	24	6	A	FORM-Name	
10	1E	30	2	-	- reserviert -	
11	20	32	3	-	- reserviert -	
12	23	35	1	B	Komponenten-Kennzeichen	2)

Länge der Zeilendrucker-Erweiterung: 36 Byte

### Anmerkung

- Die Werte für gedruckte Zeilen/Seiten können bei Druckerproblemen während des SPOOLOUTs nicht gewährleistet werden. Speziell für den Drucker vom Typ 3365 gilt, dass die Werte in folgenden Fällen ungültig sind:
  - Steuerzeichen in der Eingabedatei
  - zusätzliche Daten, die über PCL-Datei zum Drucker gesandt werden
  - der erweiterte Modus ist auf 'ON' gesetzt
  - PCL sendet spezielle Optionen zum Drucker (z.B. mehr als eine logische Seite pro physikalische Seite)

Generell gilt zudem, dass in allen den Fällen, in denen sich der Drucker nicht wie ein Zeilendrucker verhält, die Anzahl der Seiten und Zeilen nicht korrekt vervollständigt werden können.

2. Kennzeichen, ob die Ausgabe auf einem lokalen Drucker über Kanalanschluss oder auf einem RSO-Drucker über DFÜ-Verbindung erfolgte.

Mögliche Werte:

- X'01' für lokalen Drucker über Kanalanschluss
- X'02' für RSO-Drucker über DFÜ-Verbindung
- X'03' für Drucker an einem Bus-Kanalanschluss
- X'05' für Dprint-Aufträge innerhalb eines Clusters für lokalen Druckern
- X'15' für Dprint-Aufträge zwischen Clustern für lokalen Druckern
- X'16' für Dprint-Aufträge zwischen Clustern für RSO-Drucker

b) *APA-Seitendrucker-Erweiterung*

Die SPOOLOUT-Datei wurde auf einem APA-Seitendrucker ausgedruckt.

<b>Feld</b>	<b>Distanz</b>		<b>Länge</b>	<b>Format</b>	<b>Bedeutung</b>	<b>Anm.</b>
<b>Nr.</b>	<b>hex.</b>	<b>dez.</b>	<b>(Byte)</b>			
1	00	0	2	A	Erweiterungskennung 'OM'	
2	02	2	1	B	Anzahl der Elemente (X'01')	
3	03	3	1	B	Länge des Elementes (X'50')	
4	04	4	2	A	Fallkennzeichnung 'AP'	
5	06	6	2	A	mnemotechnischer Gerätename	
6	08	8	4	-	reserviert	
7	0C	12	4	-	reserviert	
8	10	16	8	A	Gerätename	
9	18	24	6	A	FORM-Name	
10	1E	30	2	-	reserviert	
11	20	32	3	-	reserviert	
12	23	35	1	B	Drucker-Zugriff	1)
13	24	36	4	B	Anzahl der Übertragungen	
14	28	40	4	B	Anzahl gedruckter Blätter	
15	2C	44	4	B	Anzahl gedruckter Seiten	
16	30	48	4	B	Verbrauchte Zeit (.01 Sek.)	
17	34	52	4	B	Anzahl gedruckter zusätzlicher Blätter	
18	38	56	4	B	Anzahl gedruckter zusätzlicher Seiten	
19	3C	60	4	B	zusätzlich verbrauchte Zeit (.01 Sek)	
20	40	64	4	B	Anzahl angeforderter PAGEDEFS	

Feld	Distanz		Länge	Format	Bedeutung	Anm.
Nr.	hex.	dez.	(Byte)			
21	44	68	4	B	Anzahl angeforderter FORMDEFs	
22	48	72	4	B	Anzahl angeforderter FONTS	
23	4C	76	4	B	Anzahl geladener FONTS	
24	50	80	4	B	Anzahl angeforderter Overlays	
25	54	84	4	B	Anzahl geladener Overlays	
26	58	88	4	B	Anzahl angeforderter Page Segments	
27	5C	92	4	B	Anzahl geladener Page Segments	
28	60	96	1	B	Flag für Eingabefach	
29	61	97	1	B	Flag für Ausgabefach	
30	62	98	1	B	Flag für Duplexdruck	
31	63	99	37	-	reserviert	

Länge der Erweiterung für APA-Seitendrucker: 136 Bytes

*Anmerkung*

1. Dieses Feld gibt an, ob es sich um einen lokalen oder verteilten Druckauftrag handelt. Also ob die Geräte über einen Kanal oder das Netzwerk angesprochen werden. Die möglichen Werte sind:
  - X'01' für einen lokalen Drucker über einen Kanal
  - X'05' Dprint-Druckaufträge innerhalb eines Clusters für lokale Drucker
  - X'15' Dprint-Druckaufträge zwischen Clustern für lokale Drucker

## 6. Erweiterung: SPOOLOUT-Dateiname

Diese Satzerweiterung ist eine Zeichenketten-Erweiterung.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'FN'	
2	02	2	1	B	X'00'	
3	03	3	1	B	Länge L des Dateinamens	
4	04	4	54	F/Z	Dateiname der SPOOLOUT-Datei	1)
5	3A	58	64	F/Z	Elementname	
6	7A	122	24	C	Elementversion	
7	92	146	8	C	Elementtyp	
8	9A	154	2	C	Anzahl der Sätze	2)

Maximale Länge der Dateinamen-Erweiterung: 156 Byte

### Anmerkung

1. Dateiname oder Name der PLAM-Bibliothek gemäß SPOOLOUT-Report. Bei Objektmoduln aus der SYSEAM-Datei enthält dieses Feld "\*EAM-OMF", bei anderen temporären Dateien aus SYSEAM enthält das Feld die zehnstellige Dateinummer. Für die katalogisierten Dateien wird der Dateiname aus dem PRINT-DOCUMENT-Kommando ausgegeben.
2. Dieses Feld entspricht den Angaben im PRINT-DOCUMENT-Kommando bezüglich der Anzahl zu druckender Sätze.

## 7. Erweiterung: Account-ID - Abrechnungskennung

Diese Erweiterung enthält Zeichenketten.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'ID'	
2	02	2	1	B	X'00'	
3	03	3	1	B	Länge L der Abrechnungskennung	
4	04	4	L	C/X	Dateiname der SPOOLOUT-Datei	1)

Länge der Erweiterung für die Abrechnungskennung: 12 Byte

*Anmerkung*

1. Die Abrechnungskennung wird vom Benutzer entweder mit dem Kommando WRITE-ACCOUNTING-RECORD (Operand USER-ACCOUNTING-STEP) oder mit dem Macro AREC (Operand ID) eingetragen. Wenn keine Abrechnungskennung für den aktuellen Druckauftrag angegeben wurde, wird das Feld mit 'XXXXXXXXXXXXXXXXXXXX' gefüllt.

**8. Erweiterung: Aufträge von externen Benutzern**

Die folgende Erweiterung ist nur gegeben, wenn der SPOOLOUT-Prozess von UNIX-Kommandos oder WPRINT-SPOOLOUT-Kommandos angestoßen wird.

Feld Nr.	Distanz		Länge (Byte)	Format	Bedeutung	Anm.
	hex.	dez.				
1	00	0	2	A	Erweiterungskennung 'EU'	
2	02	2	1	B	Anzahl der Elemente (X'00')	
3	03	3	1	B	Elementlänge	
4	04	4	L	C/X	Externe Benutzerkennung	1)

Maximallänge der Erweiterung für die Abrechnungskennung: 28 Byte

*Anmerkung*

1. Die maximale Länge der externen Benutzerkennung ist 24 Byte.

**Genereller Hinweis**

Maximale Länge des SPOOLOUT-Abrechnungssatzes: 504 Byte



## 8.5 Anwendungsregeln für die Sprache C

Spool & Print - Makros unterstützen eine Schnittstelle für C-Programmierung, um auf einige Funktionen, die ab SPOOL V4.1 vorhanden sind, zugreifen zu können. Dafür müssen aber folgende Anwendungsregeln befolgt werden.

### 8.5.1 Allgemeine Regeln

1. Der Standardheader muss mit Werten, die mit der Schnittstelle korrespondieren, initialisiert sein. Die folgende Tabelle gibt dafür die nötigen Informationen.

Makro-Name	Unit	Function	Version
STRSPPR	46	57	1
STRRSO	46	57	1
STPSPPR	46	59	1
SUPSPPR	46	60	1
SASDSPO	46	24	1

2. Zeigerfelder müssen mit ((voi \*)-1) initialisiert sein und nicht mit *null*.
3. Stringfelder müssen mit dem Leerzeichen (X'40') initialisiert sein und dürfen nicht mit \0 enden.
4. Abfragen mit Flag-Werten können für ein Feld benutzt werden. Es sollte dafür aber unbedingt die Schnittstellenbeschreibung konsultiert werden.

#### *Beispiel*

Der Exit-Operand des Makros: SASDSPO

Operandenwert	Korrespondierender Flag
EXIT=*ANY	SASDSPOexit_any
EXIT=*YES	SASDSPOexit_yes
EXIT=*NO	SASDSPOexit_no

## 5. List bzw. Wildcard-Operanden

Mit dem Eintragstyp \*LIST können Sie definieren, dass der korrespondierende Operand eine Liste enthält.

Mit dem Eintragstyp \*WILD können Sie definieren, dass die Namen teilqualifiziert mit Platzhaltern angegeben werden. Der Namensoperand ist dann nicht gesetzt und der Wildcard-Operand enthält den gegenwärtigen Wert.

\*LIST-Unterstützung in C: *flag\_cx* muss auf *flag\_list* gesetzt werden und das Tabellenelement ist mit der Liste gefüllt.

\*WILD-Unterstützung in C: *flag\_cx* muss auf *flag\_wildcard* gesetzt werden, *wldname* wird mit dem gegenwärtigen Wert gefüllt und das Tabellenelement ist mit Leerzeichen initialisiert.

## 6. Die structured Operanden FOB, PRIORITY und CHARACTER-SET benötigen eine spezielle Erwähnung, die die Initialisierung der korrespondierenden Struktur betrifft. In anderen Sprachen wird diese Initialisierung via MF-Unterstützung erledigt.

Syntax: *FIELD=(type,low,high)*

*flag set* muss benutzt werden, um den Wert des Eintragstyps zu setzen.

Im Fall eines Bereichstyps, ist der Bereich im Einklang mit folgender Tabelle zu wählen. Er muss direkt auf den gegenwärtigen Wert gesetzt werden. Die Benutzung von *flag set* ist strengstens verboten.

Operandenname		STRSPPR	SUPSPPR	SASDSPO	zu setzender Wert
FOB	low	*STD	*STD/*UNCHANGED	*MIN	0
	high	*STD	*STD/*UNCHANGED	*STD	4032
PRIORITY	from	*MIN	*MIN/*UNCHANGED	*MIN	30
	to	*MAX	*MAX/*UNCHANGED	*MAX	255
CSN	low	*MIN	*MIN/*UNCHANGED	*MIN	1
	high	*MAX	*MAX/*UNCHANGED	*MAX	64

## 8.5.2 STRSPPR

Das Feld 'specifiedX' muss auf false (=0) gesetzt werden.

## 8.5.3 SUPSPPR

1. Das RSO zugewiesene P/L-Layout korrespondiert mit dem STRSRSO-Makro.
2. Wenn im Flag-Typ 'unchanged' ausgewählt ist, muss das korrespondierende Feld 'specifiedX' auf false (=0) gesetzt werden. Dazu ist keine weitere Information für den Parameter anzufordern.
3. Das Feld 'specifiedX' muss auf true (=1) gesetzt werden, wenn der korrespondierende Wert zur Standardsetzung des Operanden in STRSPPR ausgewählt wird.
4. Das Feld für die Seitendruckerausgabe muss nicht initialisiert werden. Der Wert wird intern durch das Makro gesetzt. Das Feld existiert als ein gleiches P/L Layout für STRSPPR und SUPSPPR.
5. Werte für spezielle Flags müssen definiert werden, um die Standardwerte für die Operanden PRIORITY und CHARACTER-SET des Makros STRSPPR zu setzen. Für den Operanden PRIORITY benutzt man SUPSPPRflag\_minprio und SUPSPPRflag\_maxprio, für den Operanden CHARACTER-SET benutzt man SUPSPPRflag\_minchars und SUPSPPRflag\_maxchars.
6. Wenn RSOPAR=\*NONE ausgewählt wird, muss der Zeiger mit ((void \*) -1) initialisiert sein und nicht mit flag *none*.



---

# Literatur

## Bestellen von Handbüchern

Die Handbücher sind online unter <http://manuals.fujitsu-siemens.com> zu finden oder in gedruckter Form gegen gesondertes Entgelt unter <http://FSC-manualshop.com> zu bestellen.

**BS2000/OSD**  
**Softbooks Deutsch**  
CD-ROM

*Zielgruppe*

BS2000/OSD-Anwender

*Inhalt*

Auf der CD-ROM „BS2000/OSD SoftBooks Deutsch“ sind nahezu alle deutschen Handbücher und Readme-Dateien zur BS2000-Systemsoftware der aktuellsten BS2000/OSD-Version und auch von Vorgängerversionen gespeichert, incl. der aufgeführten Handbücher. Diese Softbooks finden Sie auch im Internet auf unserem Manual Server. Sie können in den Handbüchern nachschlagen oder sich vollständige Handbücher herunterladen.

*Bestellnummer der CD-ROM*

U26175-J8-Z125-1

*Internet-Adresse*

<http://manuals.fujitsu-siemens.com>

### **BS2000/OSD-BC**

#### **DVS-Makros**

Benutzerhandbuch

#### *Zielgruppe*

Das Handbuch wendet sich an Assembler-Programmierer.

#### *Inhalt*

Es beschreibt die Makroschnittstellen des DVS für den BS2000/OSD-Grundausbau. Es werden kurz die programmierungsrelevanten zugriffsmethoden-spezifischen Besonderheiten und anschließend die Makros in lexikalischer Form beschrieben.

### **BS2000/OSD-BC**

#### **Einführung in das DVS**

Benutzerhandbuch

#### *Zielgruppe*

Das Handbuch wendet sich an den nichtprivilegierten Anwender und an die Systembetreuung.

#### *Inhalt*

Es beschreibt die Dateiverwaltung und -verarbeitung im BS2000.

Themenschwerpunkte:

- Datenträger und Dateien
- Datei- und Katalogverwaltung
- Datei- und Datenschutz
- OPEN-, CLOSE-, EOVS-Verarbeitung
- DVS-Zugriffsmethoden (SAM, ISAM,...)

### **BS2000/OSD-BC**

#### **Einführung in die Systembetreuung**

Benutzerhandbuch

#### *Zielgruppe*

Das Handbuch wendet sich an die Systembetreuung und das Operating des Betriebssystems BS2000/OSD.

#### *Inhalt*

Es sind u.a. folgende Themen zur Verwaltung und Überwachung des BS2000/OSD-Grundausbaus enthalten: Systemeinleitung, Parameterservice, Job- und Tasksteuerung, Speicher-, Geräte-, Benutzer-, Datei-, Pubset- und Systemzeit-Verwaltung, Privilegienvergabe, Accounting und Operatorfunktionen.

**BS2000/OSD-BC**  
**Kommandos Band 1 - 5**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich sowohl an den nichtprivilegierten Anwender als auch an die Systembetreuung.

*Inhalt*

Die Bände 1 - 5 enthalten die Kommandos ADD-... bis WRITE-... (BS2000/OSD-Grundausbau und ausgewählte Produkte) mit der Funktionalität für alle Privilegien. Die Kommando- und Operandenfunktionen werden ausführlich beschrieben; viele Beispiele unterstützen das Verständnis. Am Anfang jedes Bandes informiert eine Übersicht über alle in den Bänden 1 - 5 beschriebenen Kommandos.

Der Anhang von Band 1 enthält u.a. Informationen zur Kommandoingabe, zu bedingten Jobvariablenausdrücken, Systemdateien, Auftragschaltern, Geräte- und Volumetypen. Der Anhang der Bände 4 und 5 enthält jeweils eine Übersicht zu den Ausgabespalten der SHOW-Kommandos der Komponente NDM. Der Anhang von Band 5 enthält zusätzlich eine Übersicht aller START-Kommandos.

In jedem Band ist ein umfangreiches Stichwortverzeichnis mit allen Stichwörtern der Bände 1 - 5 enthalten.

**BS2000/OSD-BC**  
**Kommandos Band 6, Ausgabe in S-Variablen und SDF-P-BASYS**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an Programmierer und Anwender, die Prozeduren erstellen.

*Inhalt*

Band 6 enthält die tabellarische Darstellung aller S-Variablen, die von den SHOW-Kommandos bei einer strukturierten Ausgabe mit Werten versorgt werden. Weitere Kapitel:

- Einführung in das Arbeiten mit S-Variablen
- SDF-P-BASYS

**BS2000/OSD**  
**Makroaufrufe an den Ablaufteil**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an alle BS2000/OSD-Assembler-Programmierer.

*Inhalt*

Das Handbuch enthält eine Zusammenstellung der Makroaufrufe an den Ablaufteil, die ausführliche Beschreibung jedes Makroaufrufs mit Hinweisen und Beispielen sowie einen ausführlichen allgemeinen Lernteil.

### **BS2000/OSD-BC**

#### **System Exits** (nur in Englisch)

User Guide

##### *Target group*

The manual addresses systems support.

##### *Contents*

The manual contains an introduction to the system exits with a description of the base mechanism. The main section comprises descriptions of all system exits, e.g. exits for the BS2000/OSD basic configuration, SPOOL, DCAM, SDF and PLAM.

### **Distributed Print Services** (BS2000/OSD)

Drucken in Computernetzen

Benutzerhandbuch

##### *Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Anwender, Geräteverwalter und die Systembetreuung des BS2000/OSD.

##### *Inhalt*

Beschrieben werden für die drei Anwendergruppen Konzept, Nutzung und Verwaltung von Distributed Print Services. Beispiele zeigen die Einsatzmöglichkeiten von Distributed Print Services.

### **DSSM/SSCM**

#### **Verwaltung von Subsystemen in BS2000/OSD**

Benutzerhandbuch

##### *Zielgruppe*

Das Handbuch wendet sich an die Systembetreuung und die Softwareberatung des BS2000/OSD.

##### *Inhalt*

Es werden das Subsystemkonzept des BS2000/OSD, die Dynamische Subsystemverwaltung DSSM und die Subsystemkatalog-Verwaltung SSCM mit den dazugehörigen Kommandos und Anweisungen beschrieben.

### **IDOM** (BS2000/OSD)

Integriertes Dokumenten und Output Management

Benutzerhandbuch

##### *Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Anwender und die Systembetreuung des BS2000/OSD.

##### *Inhalt*

Das Handbuch beschreibt das SPOOL-Subsystem IDOM, das ein Dokumenten- und Output-Management im BS2000/OSD ermöglicht.



**JV (BS2000/OSD)****Jobvariablen**

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich sowohl an den nichtprivilegierten Anwender als auch an die Systembetreuung.

*Inhalt*

Es beschreibt die Verwaltung und die verschiedenen Einsatzmöglichkeiten von Jobvariablen. Die Kommandobeschreibungen sind getrennt nach den Funktionsbereichen der JVs aufgeführt. Die Makroaufrufe sind in einem eigenen Kapitel beschrieben.

**PRISMAproduction/BS2000****SPS**

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Anwender und die Systembetreuung des BS2000/OSD.

*Inhalt*

Das Handbuch beschreibt das Drucken auf APA-Druckern mit Hilfe des SPOOL-Subsystems SPS. Es werden folgende Themenbereiche behandelt: die Installation von SPS und APA-Druckern, das Erstellen und Bereitstellen von Druckdatei und Druckressourcen, die Migration von HP/HP90-Anwendungen in die AFP/SPDS-Welt, Parameterdateien, die Möglichkeiten des Druckens und der Druckgestaltung und das Verhalten in Fehlerfällen.

**PRM (BS2000/OSD)**

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an SPOOL-Anwender, Systembetreuung und RSO-Geräteverwalter.

*Inhalt*

Das Handbuch beschreibt das Dienstprogramm PRM zur Erstellung und Verwaltung von Druckressourcen für BS2000-SPOOL. Es werden die zwei Benutzeroberflächen des Dienstprogramms beschrieben: die SDF-Anweisungen für Dialog- und Stapelbetrieb und die FHS-basierte Menüoberfläche für den Dialogbetrieb.

**RSO (BS2000/OSD)**  
**Remote SPOOL Output**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Benutzer, RSO-Geräteverwalter, SPOOL-Verwalter und Systembetreuer des BS2000/OSD.

*Inhalt*

Für die einzelnen Anwendergruppen werden die Aufgaben und Möglichkeiten zur Nutzung und Steuerung von dezentralen Druckern (RSO-Drucker) beschrieben. Das Handbuch enthält die Druckermerkmale aller RSO-Drucker.

**SDF (BS2000/OSD)**  
**Einführung in die Dialogschnittstelle SDF**  
Benutzerhandbuch

*Zielgruppe*

BS2000/OSD-Anwender

*Inhalt*

Das Handbuch beschreibt die Dialog-Eingabe von Kommandos und Anweisungen im SDF-Format. Ein Schnelleinstieg mit leicht nachvollziehbaren Beispielen und weitere umfangreiche Beispiele erleichtern die Anwendung. SDF-Syntaxdateien werden erklärt.

**SDF-P (BS2000/OSD)**  
**Programmieren in der Kommandosprache**  
Benutzerhandbuch

*Zielgruppe*

BS2000-Anwender und Systembetreuung.

*Inhalt*

SDF-P ist eine strukturierte Prozedursprache im BS2000. Nach einführenden Kapiteln zum Prozedur- und Variablenkonzept werden Kommandos, Funktionen und Makros ausführlich beschrieben.

**SNS (BS2000/OSD)**  
**SPOOL Notification Service**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Anwender und die Systembetreuung des BS2000/OSD.

*Inhalt*

Das Handbuch beschreibt das Subsystem SNS, ein Werkzeug zum Senden und Verwalten von Benachrichtigungen im Rahmen von BS2000/OSD.

**SPCONV (BS2000/OSD)**

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an Systembetreuung und Geräteverwalter.

*Inhalt*

Das Handbuch beschreibt das Subsystem SPCONV. Das Handbuch beschreibt den Aufbau des Filtersystems und den Einsatz von Filtern.

**SPOOL (BS2000/OSD)**

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Anwender, den Spool & Print - Verwalter, den RSO-Geräteverwalter und die Systembetreuung.

*Inhalt*

Es wird der Betrieb von SPOOL beschrieben.

**Spool & Print - Kommandos (BS2000/OSD)**

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Anwender, den Geräteverwalter, den Cluster-Verwalter, den SPOOL-Verwalter und die Systembetreuung.

*Inhalt*

Es werden die verfügbaren Kommandos von SPOOL, Dprint, RSO und SPS beschrieben. Kommandos zur Subsystemverwaltung und zur Auftragssteuerung sind nicht beschrieben.

**Spool & Print - Meldungen**

Benutzerhandbuch

*Zielgruppe*

Systembetreuung, RSO-Geräteverwalter und nichtprivilegierte Benutzer.

*Inhalt*

Meldungen zu SPOOL, RSO, SPSEIVE, PRM, SPCONV, SPS, SPOOLSYS, IDOM und Distributed Print Services. Meldungstexte in Deutsch und Englisch, Bedeutungs- und Maßnahmetexte in Deutsch. Garantierte Meldungen sind gekennzeichnet.

### **SPSERVE (BS2000/OSD)**

Benutzerhandbuch

#### *Zielgruppe*

Das Handbuch wendet sich an nichtprivilegierte Benutzer, RSO-Geräteverwalter, Dprint-Cluster-Verwalter und die Systembedienung des BS2000/OSD.

#### *Inhalt*

Es wird das Dienstprogramm SPSEVERE mit allen Anweisungen dargestellt. Dabei werden alle Ergänzungen von SPOOL, RSO, SPCONV, PRM, Distributed Print Services und SPS berücksichtigt.

### **Wprint (Windows)**

#### *Zielgruppe*

Anwender, die aus Windows-Anwendungen heraus drucken möchten und Systemverwalter von SINIX, UNIX-Systemen und BS2000/OSD Systemen bzw. SINIX-Spool-Verwalter.

#### *Inhalt*

Das Handbuch beschreibt die Funktionsweise und den Betrieb von Wprint-Server und Wprint-Client, sowie die Installation und Konfiguration dieser beiden Komponenten.

### **Xprint (Reliant UNIX-Systeme)**

#### **Application Programming Interface (API)**

#### *Zielgruppe*

Programmierer, die die Xprint-Funktionalität in ihren Programmen nutzen wollen.

#### *Inhalt*

Das Buch enthält eine Referenz aller Xprint API-Funktionen und der zugehörigen Makros. Es beschreibt, wie die Funktionen des Xprint API in eigenen Programmen genutzt werden können.

### **Xprint (Reliant UNIX-Systeme)**

#### **Leitfaden für Benutzer und Verwalter**

Benutzerhandbuch

#### *Zielgruppe*

Benutzer von Reliant UNIX-Systemen bzw. Systemverwalter, Xprint-Verwalter

#### *Inhalt*

Das Handbuch gibt Informationen zum Xprint-Konzept und dokumentiert, aus welchem Objekten Xprint besteht. Zudem wird die Installation von Xprint sowie die Bedienung über die Kommandooberfläche beschrieben.

**Xprint (Reliant UNIX-Systeme)****Menüs**

Benutzerhandbuch

*Zielgruppe*

Benutzer von Reliant UNIX-Systemen bzw. Systemverwalter, Xprint-Verwalter

*Inhalt*

Das Handbuch beschreibt die Bedienung von Xprint über die Menüoberfläche.

**Xprint (Reliant UNIX-Systeme)**

Referenzhandbuch

*Zielgruppe*

Benutzer von Reliant UNIX-Systemen bzw. Systemverwalter, Xprint-Verwalter

*Inhalt*

Das Handbuch beschreibt die Xprint-Kommandos in alphabetischer Reihenfolge, dokumentiert die Xprint-Meldungen, die Konfigurationsdateien für Xprint-Objekte sowie die Standard-Dateinformate von Xprint und gibt einen Überblick über die Arbeitsweise des Xprint-Systems.

Das Handbuch vermittelt grundlegende Kenntnisse über das Xprint-System und gibt anhand von ausgewählten Beispielen konkrete Handlungsanweisungen.



---

# Stichwörter

\$DPFL-Parameterliste  
Exit 093 553

## A

Abrechnungssatz  
SPLO 613

Accounting-Satz  
SPLO 613

ADD-DPRINT-FT-LIMIT (Kommando) 595

Anforderungen  
Mindest-, an Drucker 30

Anwendungshinweise  
Exit 097 574

Anwendungsregeln für C 625

APA-Drucker  
Format-Definition 189

ASA  
Drucksteuerbyte 33  
Vorschubsteuerzeichen 33

Aufrufformat  
Makro CANJOB 215  
Makro MODJRSO 357  
Makro MODPJAT 220  
Makro MPJADPC 364  
Makro MPJASPS 366  
Makro PRNTDOC 44  
Makro PRNTDPC 182  
Makro PRNTRSO 10, 176  
Makro PRNTSPS 187  
Makro SHOPJAT 297  
Makro SHOPJST 305  
Makro SJAODOF 372  
Makro SJAODOP 374  
Makro SJAOLCT 377

Makro SJAOPJC 380  
Makro SJAORSD 381  
Makro SJAORSO 384  
Makro SJAOTOP 383  
Makro SJSODES 391, 394, 397  
Makro SJSODIS 400  
Makro SJSODSS 402  
Makro SJSOORI 404  
Makro SJSOORS 406  
Makro SJSOTRA 407  
Makro SJSOTRS 409  
Makro SPJADPC 385  
Makro SPJASPS 386  
Makro SPJRSO 387  
Makro SPSINF 200, 201

Ausgabe  
auf Drucker 29  
Aufbau der Ausgabezeile sedezimal 86, 254  
Datei mit Makro PRNTDOC 29  
Vorschubsteuerzeichen 32

Ausgabe-Routine  
Exit 095 567

Ausgeben  
SPOOL-Parameter 199

Auswerten  
Steuerzeichen 36

## B

Bar Code 9, 197  
Bibliothekselement auf Drucker ausgeben 29

**C**

- CANPJOB (Makro) 215
  - Parameterliste 219
  - Returncode 218
- C-Headerfile (PRNTDPC-Makro) 185

**D**

- Datei ausgeben
  - auf Drucker 29
- Deckblatt
  - Aufbau 543
  - Exit 091 535
- Definieren
  - Formular 37
  - Loop 31, 37
  - Zeichensatz 37
- Dienstprogramme
  - zur Verwaltung 2
- Doppelseitiger Druck 109
- Dprint
  - Kurzbeschreibung 1
- Druckauftrag steuern 29
- Drucken
  - Formular angeben 31
  - Papieranschub steuern 31
- Drucker
  - Aufbau der Ausgabezeile sedezimal 86, 254
  - Ausgabe auf 29
  - auswählen 30
  - Vorschubsteuerzeichen 32
- Druckerausgabe
  - Exit 095 567
- Druckerauswahl
  - implizite 30
- Druckersteuerbyte 32
- Druckseitengestaltung
  - Font 189
- Drucksteuerbyte
  - ASA 33
  - IBM 32
- DSECT (Makro SSVSCB) 524

**E**

- EBCDIC-Drucksteuerbyte 32
- EBCDIC-Vorschubsteuerzeichen 32
- Einseitiger Druck 109
- Exit
  - 090 für SPOOLOUT 527
  - 091 für Deckblatt 535
  - 092 für Schlussblatt 535
  - 093 (Kunden-Filter) 550
  - 094 für Ressourcen-Konvertierung 563
  - 095 Ausgabe-Routine 567
  - 095 für Druckerausgabe 567
  - 096 für Server-Auswahl 570
  - 097 (File Transfer steuern) 572
  - Allgemeines 524
  - für Deckblatt (091) 535
  - für Schlussblatt (092) 535
- Exit 090
  - Exit-Routine 527
  - Parameterliste 528
  - Programmierbeispiel 532
  - Returncode 527
- Exit 091
  - Parameterliste 537
  - Programmierbeispiel 546
  - Returncode 536
- Exit 092
  - Parameterliste 540
  - Programmierbeispiel 548
- Exit 093
  - \$DPFL-Parameterliste 553
  - Programmierbeispiel 557
  - Returncode 552
  - Sicherheitsaspekte 562
- Exit 094
  - Parameterliste 564
  - Programmierbeispiel 565
  - Returncode 564
- Exit 095
  - Parameterliste 568
  - Returncode 567
- Exit 096
  - Parameterliste 571
  - Returncode 570



- Exit 097
  - Anwendungshinweise 574
  - Parameterliste 576
  - Programmierbeispiel 577
  - Returncode 575
- Exits 523
- F**
- Fehlerbehandlung bei virtuellen Druckern 515
- Fehlerverhalten
  - Jobvariable 38
- File Transfer steuern
  - Exit 097 572
- Font
  - zur Druckseitengestaltung 189
- Format-Definition
  - APA-Drucker 189
- Formular
  - angeben zum Drucken 31
  - definieren 37
- I**
- IBM-Druckersteuerbyte 32
- IDOM 2
- Informationen
  - über SPOOL-Parameter 199
- Inhalt
  - Jobvariable für SPOOLOUT 38
- J**
- Jobvariable
  - für SPOOLOUT 37
  - Inhalt 38
  - SPOOL-Auftrag 37
  - SPOOLOUT, Fehlerverhalten 38
- K**
- Kommando
  - ADD-DPRINT-FT-LIMIT 595
  - MODIFY-DPRINT-FT-LIMIT 594
  - REMOVE-DPRINT-FT-LIMIT 596
  - SHOW-DPRINT-FT-LIMIT 593
- Kunden-Filter
  - Exit 093 550
- L**
- Langform
  - Operandennamen 342
- Laserdrucker-Beispiele 35
- Layout-Steuerzeichen
  - auswerten 36
- Loop
  - definieren 31, 37
  - Papiervorschub steuern 31
- LP65-Drucker
  - Papierausgabefach 88, 263
  - Papiereingabefach 70, 257
- M**
- Makro
  - CANPJOB 215
  - MODJRSO 357
  - MODPJAT 220
  - MPJADPC 364
  - MPJASPS 366
  - OLTPRQX 10
  - PRNTDOC 29
  - PRNTDPC 182
  - PRNTRSO 10, 176
  - PRNTSPS 187
  - SHOPJAT 297
  - SHOPJST 305
  - SJAODOF 372
  - SJAODOP 374
  - SJAOLCT 377
  - SJAOPJC 380
  - SJAORSO 381
  - SJAOTOP 383
  - SJSODES 391, 394, 397
  - SJSODIS 400
  - SJSODSS 402
  - SJSOORI 404
  - SJSOORS 406
  - SJSOTRA 407
  - SJSOTRS 409
  - SPJADPC 385
  - SPJARSO 384
  - SPJASPS 386
  - SPJSRSO 387

## Makro (Forts.)

- SPSINF 199
- SPVDEND 509
- SPVDINI 503
- SPVDRCV 505
- SPVDRET 507
- SSVSCB 524

## Makroaufruf

- Typen 602

## Makroaufrufe

- Metasyntax 598, 600
- Standardheader 606
- Typ O 602
- Typ R 602

## Makroaufrufformat 597

## Makrosyntax 600

## Makrotyp

- allgemeine Beschreibung 602
- Aufrufformat 602
- O-Typ 602
- R-Typ 602

## Metasyntax des Makros 598, 599

## Mindestanforderungen

- an Drucker zum Ausdrucken 30

## MODIFY-DPRINT-FT-LIMIT (Kommando) 594

## MODJAT (Makro)

- Returncode 285

## MODJRSO (Makro) 357

- Parameterliste 362

## MODPJAT (Makro) 220

- Parameterliste 286

## Monitorvariable

- für SPOOLOUT 37

## Monitorvariable (SPOOL-Auftrag) 37

## MONJV (SPOOL-Auftrag) 37

## MPJADPC (Makro) 364

- Parameterliste 365

## MPJASPS (Makro) 366

- Parameterliste 370

## O

## OLTPRQX 10

## Operanden

- Unverträglichkeit, Makro PRNTDOC 115

## Operandennamen

- Langform 342

## P

## Papierausgabefach 88, 263

- LP65-Drucker 88, 263

- RSO-Drucker 88, 263

- Sortiermechanismus 89, 104, 264, 278

## Papiereingabefach 70, 257

- LP65-Drucker 70, 257

- RSO-Drucker 70, 257

## Papiervorschub steuern 31

## Parameterliste

- CANPJOB (Makro) 219

- Exit 090 528

- Exit 091 537

- Exit 092 540

- Exit 093 553

- Exit 094 564

- Exit 095 568

- Exit 096 571

- Exit 097 576

- MODJRSO (Makro) 362

- MODPJAT (Makro) 286

- MPJADPC (Makro) 365

- MPJASPS (Makro) 370

- PRNTDOC (Makro) 119

- PRNTDPC (Makro) 184

- PRNTRSO (Makro) 181

- PRNTSPS (Makro) 191

- SHOPJAT (Makro) 302

- SHOPJST (Makro) 347, 436, 454

- SJAODOF (Makro) 372

- SJAODOP (Makro) 374

- SJAOLCT (Makro) 377

- SJAOPJC (Makro) 380

- SJAORSO (Makro) 381

- SJAOTOP (Makro) 383

- SJSODES (Makro) 392, 395, 398

- SJSODIS (Makro) 401

- SJSODSS (Makro) 403

- SJSOORI (Makro) 404

- SJSOORS (Makro) 406

- SJSOTRA (Makro) 407

- Parameterliste (Forts.)
  - SJSOTRS (Makro) [409](#)
  - SPJADPC (Makro) [385](#)
  - SPJARSO (Makro) [384](#)
  - SPJASPS (Makro) [386](#)
  - SPJSRSO (Makro) [388](#)
- PCL-Datei [188](#)
- Privat
  - Formular definieren [37](#)
  - Loop [31](#)
  - Loop definieren [37](#)
  - Zeichensatz definieren [37](#)
- PRM
  - Kurzbeschreibung [2](#)
- PRNTDOC (Makro) [29](#)
  - Parameterliste [119](#)
  - Returncode [117](#)
  - unverträgliche Operanden [115](#)
- PRNTDPC (Makro) [182](#)
  - Parameterliste [184](#)
- PRNTRSO (Makro) [10, 176](#)
  - Parameterliste [181](#)
- PRNTSPS (Makro) [187](#)
  - Parameterliste [191](#)
- Programmierbeispiel
  - Exit 090 [532](#)
  - Exit 091 [546](#)
  - Exit 092 [548](#)
  - Exit 093 [557](#)
  - Exit 094 [565](#)
  - Exit 097 [577](#)
- R**
- REMOVE-DPRINT-FT-LIMIT (Kommando) [596](#)
- Ressourcen-Konvertierung
  - Exit 094 [563](#)
- Returncode
  - Exit 090 [527](#)
  - Exit 091 [536](#)
  - Exit 093 [552](#)
  - Exit 094 [564](#)
  - Exit 095 [567](#)
  - Exit 096 [570](#)
  - Exit 097 [575](#)
- Makro CANPJOB [218](#)
- Makro MODPJAT [285](#)
- Makro PRNTDOC [117](#)
- Makro SHOPJAT [301](#)
- Makro SHOPJST [345](#)
- Makro SPSINF [206](#)
- Rotation (Seitendrehung) [100, 256](#)
- RSO
  - Kurzbeschreibung [1](#)
- RSO-Drucker
  - Papierausgabefach [88, 263](#)
  - Papiereingabefach [70, 257](#)
- S**
- SASDSPO (Makro) [412](#)
- SCB [524](#)
- Schlussblatt
  - Aufbau [543](#)
  - Exit 092 [535](#)
- Schlussblatt-Exit (092) [535](#)
- Seiten-Definition [188](#)
- Seitendrehung [100, 256](#)
- Server-Auswahl
  - Exit 096 [570](#)
- SHOPJAT (Makro) [297](#)
  - Parameterliste [302](#)
  - Returncode [301](#)
- SHOPJST (Makro) [305](#)
  - Parameterliste [347, 436, 454](#)
  - Returncode [345](#)
- SHOW-DPRINT-FT-LIMIT (Kommando) [593](#)
- Sicherheitsaspekte Exit 093 [562](#)
- Siemens
  - Druckersteuerbyte [32](#)
  - Vorschubsteuerzeichen [33](#)
- SJAODOF (Makro) [372](#)
  - Parameterliste [372](#)
- SJAODOP (Makro) [374](#)
  - Parameterliste [374](#)
- SJAOLCT (Makro) [377](#)
  - Parameterliste [377](#)
- SJAOPJC (Makro) [380](#)
  - Parameterliste [380](#)

- SJAORSD (Makro) [381](#)
    - Parameterliste [381](#)
  - SJAOTOP (Makro) [383](#)
    - Parameterliste [383](#)
  - SJSODES (Makro) [391](#), [394](#), [397](#)
    - Parameterliste [392](#), [395](#), [398](#)
  - SJSODIS (Makro) [400](#)
    - Parameterliste [401](#)
  - SJSODSS (Makro) [402](#)
    - Parameterliste [403](#)
  - SJSOORI (Makro) [404](#)
    - Parameterliste [404](#)
  - SJSOORS (Makro) [406](#)
    - Parameterliste [406](#)
  - SJSOTRA (Makro) [407](#)
    - Parameterliste [407](#)
  - SJSOTRS (Makro) [409](#)
    - Parameterliste [409](#)
  - SNPPRNT (Makro) [192](#)
  - SNS
    - Kurzbeschreibung [2](#)
  - Sortiermechanismus
    - Papierausgabefach [89](#), [104](#), [264](#), [278](#)
  - SPCONV
    - Kurzbeschreibung [2](#)
  - SPJADPC (Makro) [385](#)
    - Parameterliste [385](#)
  - SPJARSO (Makro) [384](#)
    - Parameterliste [384](#)
  - SPJASPS (Makro) [386](#)
    - Parameterliste [386](#)
  - SPJSRSO (Makro) [387](#)
    - Parameterliste [388](#)
  - SPLO [614](#)
  - SPOOL
    - Exits [523](#)
    - Kurzbeschreibung [1](#)
    - Parameterdatei, Inhalt ausgeben [199](#)
  - Spool & Print Services
    - Kurzbeschreibung [1](#)
  - Spool&Print Explorer
    - Kurzbeschreibung [2](#)
  - SPOOL-Accounting-Satz
    - SPLO [613](#)
  - SPOOL-Control-Block [29](#)
  - SPOOL-Jobvariable [37](#)
  - SPOOLOUT
    - Exit 090 [527](#)
    - Fehlerverhalten der Jobvariable [38](#)
    - Inhalt der Jobvariable [38](#)
    - steuern [30](#)
    - überwachen [37](#)
    - Warteschlange [29](#)
  - SPOOLOUT-Abrechnungssatz
    - SPLO [613](#)
  - SPOOLOUT-Exit (090) [527](#)
  - SPS
    - Kurzbeschreibung [1](#)
  - SPSERVE
    - Kurzbeschreibung [2](#)
  - SPSINF (Makro) [199](#)
    - Returncode [206](#)
  - SPVDEND (Makro) [509](#)
  - SPVDINI (Makro) [503](#)
  - SPVDRCV (Makro) [505](#)
  - SPVDRET (Makro) [507](#)
  - SSVBARC (macro) [197](#)
  - SSVSCB (Makro) [524](#)
  - Standard-Deckblatt [545](#)
  - Standardheader [606](#)
  - Steuerzeichen auswerten [36](#)
  - STPSPPR (Makro) [452](#)
  - STRSPPR (Makro) [457](#)
  - Struktur-Layouts [15](#)
  - SUPSPPR (Makro) [478](#)
  - Systemdeckblatt [544](#)
- ## T
- TRC-Steuerzeichen [189](#), [368](#)
- ## U
- Überwachen
    - SPOOLOUT [37](#)
  - Unverträgliche Operanden
    - Makro PRNTDOC [115](#)

**V**

## Variable

- Fehlerverhalten der SPOOLOUT-  
Jobvariable [38](#)
- Inhalt bei SPOOLOUT-Überwachung [38](#)
- SPOOLOUT überwachen [37](#)

## Verwaltung

- Dienstprogramme [2](#)

## virtueller Drucker

- Druckaufträge verarbeiten [513](#)
- Fehlerbehandlung [515](#)
- initialisieren des Dialogs [511](#)
- Kommunikation mit Anwendung [511](#)
- PRNTDOC (Makro) [111](#)
- SPVDEND (Makro) [509](#)
- SPVDINI (Makro) [503](#)
- SPVDRCV (Makro) [505](#)
- SPVDRET (Makro) [507](#)
- Verarbeitung stoppen [514](#)

## Vorschubsteuerzeichen

- für die Ausgabe [32](#)
- Siemens [33](#)

**W**

## Warteschlange

- SPOOLOUT- [29](#)

## Werte festlegen

- für MODPJAT-Makro [357](#)
- für PRNTDOC-Makro [10](#), [176](#)
- für SHOPJAT-Makro [384](#)
- für SHOPJST-Makro [387](#)

**Z**Zeichensatz [33](#)

- angeben [33](#)
- definieren [37](#)
- Pool [34](#)

## Zuordnung

- Zeichensatz [33](#)



---

# Inhalt

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Kurzbeschreibung der Spool & Print Services	1
1.2	Zielgruppe des Handbuchs	4
1.3	Aufbau des Handbuchs	4
1.4	Änderungen der Makros und Exits gegenüber den Vorgängerversionen	5
1.5	Verwendete Darstellungsmittel	6
1.6	Readme-Datei	7
<b>2</b>	<b>Makros für die Ausgabe von Dateien</b>	<b>9</b>
	OLTPRQX - UTM-Druckparameter festlegen	10
	PRNTDOC (für SPOOL, SPS, RSO und Dprint) - Datei ausgeben	29
	SPOOLOUT steuern	30
	Papieranschub steuern	31
	Unverträgliche Operanden im Makro PRNTDOC	115
	Returncodes	117
	Datenbeschreibung und Beispiel für die Programmiersprache Assembler	119
2.1	Die C-Schnittstelle für die Funktionen des Makros PRNTDOC	136
2.2	Die COBOL-Schnittstelle für die Funktionen des Makros PRNTDOC	152
2.3	Unterstützende Makros für PRNTDOC	175
	PRNTRSO - RSO-Parameter für PRNTDOC-Makro festlegen	176
	PRNTDPC - Dprint-Parameter für PRNTDOC-Makro definieren	182
	PRNTSPS - SPS-APA-Parameter für PRNTDOC-Makro definieren	187
	SNPPRNT - Notification-Parameter für PRNTDOC definieren	192
	SSVBARC - Datenkonvertierung für Bar Codes	197
<b>3</b>	<b>Makros für die Ausgabe von Informationen</b>	<b>199</b>
	SPSINF - Informationen anfordern	199
<b>4</b>	<b>Makros für die Verwaltung von Druckaufträgen</b>	<b>213</b>
	CANPJOB - Druckauftrag abbrechen	215
	MODPJAT - Attribute eines Druckauftrags modifizieren	220
	SHOPJAT - Attribute eines Druckauftrags anzeigen	297
	SHOPJST - Status eines Druckauftrags anzeigen	305

4.1	Unterstützende Makros für MODPJAT .....	356
	MODJRSO - RSO-Parameter für MODPJAT-Makro festlegen .....	357
	MPJADPC - Dprint-Parameter für MODPJAT-Makro definieren .....	364
	MPJASPS - SPS-Parameter für MODPJAT-Makro definieren .....	366
4.2	Ausgabestrukturen des SHOPJAT-Makros .....	371
	SJAODOF - Ausgabestruktur für INFO=*DOCUMENT_FORMAT .....	372
	SJAODOP - Ausgabestruktur für INFO=*DOCUMENT_PART .....	374
	SJAOLCT - Ausgabestruktur für INFO=*LAYOUT_CONTROL .....	377
	SJAOPJC - Ausgabestruktur für INFO=*PRINT_JOB_CONTROL .....	380
	SJAORSD - Ausgabestruktur für INFO=*RESOURCES_DESCRIPTION .....	381
	SJAOTOP - Ausgabestruktur für INFO=*TO_PRINTER .....	383
	SPJARSO - RSO-Ausgabestruktur für SHOPJAT-Makro .....	384
	SPJADPC - Dprint-Ausgabestruktur für SHOPJAT-Makro .....	385
	SPJASPS - SPS-Ausgabestruktur für SHOPJAT-Makro .....	386
4.3	Unterstützendes Makro des SHOPJST-Makros .....	387
	SPJSRSO - RSO-Parameter für SHOPJST-Makro festlegen .....	387
4.4	Ausgabestrukturen des SHOPJST-Makros .....	390
	SJSODES - Ausgabestruktur für DESTINATION .....	391
	SJSODES - Ausgabestruktur für SPOOL-FILTER .....	394
	SJSODES - Ausgabestruktur für RSO-FILTER .....	397
	SJSODIS - Ausgabestruktur für DISTRIBUTED .....	400
	SJSODSS - Ausgabestruktur für DESTINATION und UNIX-Systeme .....	402
	SJSOORI - Ausgabestruktur für ORIGIN .....	404
	SJSOORS - Ausgabestruktur für ORIGIN und UNIX-Systeme .....	406
	SJSOTRA - Ausgabestruktur für TRAITS .....	407
	SJSOTRS - Ausgabestruktur für TRAITS und UNIX-Systeme .....	409
<b>5</b>	<b>Makros für die Verwaltung von Druckern .....</b>	<b>411</b>
	SASDSPO - Informationen über aktive SPOOL-Geräte anfordern .....	412
	STPSPPR - Druckerzuweisung aufheben .....	452
	STRSPPR - Druckern Druckaufträge zuweisen .....	457
	STRSRSO - Drucker mit RSO-spezifischen Parametern zuweisen .....	475
	SUPSPPR - Verwaltungsparameter für Drucker ändern .....	478
<b>6</b>	<b>Makros für virtuelle Drucker .....</b>	<b>499</b>
6.1	Einsatz virtueller Drucker im synchronen Betrieb .....	500
6.2	Programmschnittstelle für den synchronen Betrieb .....	502
	SPVDINI - Dialog mit virtuellem Drucker initialisieren .....	503
	SPVDRCV - Anwendungstask empfangsbereit setzen .....	505
	SPVDRET - Informationen über den aktuellen Auftrag zum virtuellen Drucker senden .....	507
	SPVDEND - Dialogverbindung beenden .....	509



6.3	Kommunikation zwischen virtuellem Drucker und Anwendung	511
6.3.1	Initialisieren des Dialogs	511
6.3.2	Druckaufträge verarbeiten	513
6.3.3	Verarbeitung stoppen	514
6.3.4	Fehlerbehandlung	515
6.3.5	Wiederaufsetzen nach Fehlern	515
6.4	Beispiel: Kommunikation mit einem virtuellen Drucker im synchronen Betrieb	516
<b>7</b>	<b>Spool-Exits</b>	<b>523</b>
7.1	Allgemeines zu den Spool-Exits	524
7.2	SPOOLOUT-Exit (090)	527
7.3	Exits für Deckblatt und Schlussblatt (091/092)	535
7.4	Exit für Kunden-Filter (093)	550
7.5	Ressourcen-Routine (094)	563
7.6	Ausgabe-Routine (095)	567
7.7	Exit für Server-Auswahl (096)	570
7.8	Exit zum Steuern des File Transfers für Dprint-Aufträge (097)	572
7.9	Unterstützende Kommandos für den Exit 097	593
	SHOW-DPRINT-FT-LIMIT	593
	MODIFY-DPRINT-FT-LIMIT	594
	ADD-DPRINT-FT-LIMIT	595
	REMOVE-DPRINT-FT-LIMIT	596
<b>8</b>	<b>Anhang</b>	<b>597</b>
8.1	Makrosyntax	597
	Makroaufrufformat	597
	„Neue“ Metasyntax der Makroaufrufformate	598
	„Alte“ Metasyntax der Makroaufrufformate	600
	Typen von Makroaufrufen	602
	Standardheader	606
	Musterzeichen	608
8.2	Funktionsweise des Exit-Mechanismus	609
8.3	Schlussblatt für lokalen SPOOL	611
8.4	SPOOLOUT-Abrechnungssatz SPLO	613
8.5	Anwendungsregeln für die Sprache C	625
8.5.1	Allgemeine Regeln	625
8.5.2	STRSPPR	627
8.5.3	SUPSPPR	627
	<b>Literatur</b>	<b>629</b>
	<b>Stichwörter</b>	<b>639</b>



---

# Spool & Print - Makros und Exits (BS2000/OSD)

## Benutzerhandbuch

### *Zielgruppe*

Das Handbuch wendet sich an Programmierer, die die Spool & Print Services in ihren Programmen direkt ansprechen wollen.

### *Inhalt*

Das Handbuch beschreibt die Makros und Exits der Spool & Print Services einschließlich der Makros für virtuelle Drucker. Die Beschreibung der Makros ist nach Funktionen geordnet.

Stand der Beschreibung:

- Distributed Print Services V1.1B
- RSO V3.4A
- SPCONV V1.0A
- SPOOL V4.6
- SPS V3.8
- SPSEERVE V2.8

**Ausgabe: Mai 2005**

**Datei: SPMAKROS.PDF**

Copyright © Fujitsu Siemens Computers GmbH, 2005.

Alle Rechte vorbehalten.

Liefermöglichkeiten und technische Änderungen vorbehalten.

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.

Dieses Handbuch wurde erstellt von  
cognitas. Gesellschaft für Technik-Dokumentation mbH  
[www.cognitas.de](http://www.cognitas.de)

Fujitsu Siemens Computers GmbH  
Handbuchredaktion  
81730 München

# Kritik Anregungen Korrekturen

**Fax: 0 700 / 372 00001**

e-mail: [manuals@fujitsu-siemens.com](mailto:manuals@fujitsu-siemens.com)  
<http://manuals.fujitsu-siemens.com>

---

Absender

---

Kommentar zu Spool & Print  
Makros und Exits



## Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format ...@[ts.fujitsu.com](mailto:ts.fujitsu.com).

The Internet pages of Fujitsu Technology Solutions are available at [http://ts.fujitsu.com/...](http://ts.fujitsu.com/) and the user documentation at <http://manuals.ts.fujitsu.com>.

Copyright Fujitsu Technology Solutions, 2009

## Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form ...@[ts.fujitsu.com](mailto:ts.fujitsu.com).

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter [http://de.ts.fujitsu.com/...](http://de.ts.fujitsu.com/), und unter <http://manuals.ts.fujitsu.com> finden Sie die Benutzerdokumentation.

Copyright Fujitsu Technology Solutions, 2009